Hindawi
Journal of Mathematics
Volume 2020, Article ID 2412491, 10 pages
https://doi.org/10.1155/2020/2412491

Research Article

Ninety-Six Distinct Real Matrices for Representing a Quaternion Number

W. E. Ahmed [1,2]

[1] *Mathematics and Statistics Department, Faculty of Science, Imam Mohammad IBN Saud Islamic University, Riyadh, Saudi Arabia*
[2] *Department of Basics and Engineering Sciences, Faculty of Engineering, University of Khartoum, Khartoum, Sudan*

Correspondence should be addressed to W. E. Ahmed; waahmed@imamu.edu.sa

Received 24 June 2020; Revised 30 September 2020; Accepted 28 October 2020; Published 4 December 2020

Academic Editor: Frank Uhlig

In this paper, we investigate on the number of all possible real matrices representing a quaternion number as three 4×4 skew-symmetric matrices plus the identity matrix of order 4, and how to determine these matrices. We establish that there are 96 distinct real matrices having this property, and by matrix row operations, we obtain these matrices.

1. Introduction

Any quaternion number can be expressed in the form $a_0 + a_1 i_1 + a_2 i_2 + a_3 i_3$, where a_0, a_1, a_2, and a_3 are real numbers, and i_1, i_2, and i_3 are imaginary units which hold Hamilton's rule:

$$\begin{aligned} i_1 i_2 i_3 &= -1, \\ i_1^2 = i_2^2 &= i_3^2 = -1. \end{aligned} \tag{1}$$

Table 1 gives all multiplications of these imaginary units. The quaternion multiplication is not commutative.

Quaternions have a great impact in developing mathematics [1, 2], for a review and to find out how quaternions were discovered by Sir William Rowan Hamilton, see [3–5] and [6] to take a general view on his life.

Quaternion numbers have representation as four-dimensional matrices over the real numbers [7], to see this, let

$$a = a_0 + a_1 i_1 + a_2 i_2 + a_3 i_3. \tag{2}$$

Multiply (2) from the right by i_1, i_2, and i_3, respectively,

$$\begin{aligned} a \cdot i_1 &= a_0 i_1 - a_1 - a_2 i_3 + a_3 i_2, \\ a \cdot i_2 &= a_0 i_2 + a_1 i_3 - a_2 - a_3 i_1, \\ a \cdot i_3 &= a_0 i_3 - a_1 i_2 + a_2 i_1 - a_3. \end{aligned} \tag{3}$$

We can make a matrix of the coefficients using the basis $\{1, i_1, i_2, i_3\}$ as follows:

$$\begin{aligned} A_1 &= \begin{bmatrix} a_0 & -a_1 & -a_2 & -a_3 \\ a_1 & a_0 & -a_3 & a_2 \\ a_2 & a_3 & a_0 & -a_1 \\ a_3 & -a_2 & a_1 & a_0 \end{bmatrix} = a_0 \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} + a_1 \begin{bmatrix} 0 & -1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 \\ 0 & 0 & 1 & 0 \end{bmatrix} \\ &+ a_2 \begin{bmatrix} 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \end{bmatrix} + a_3 \begin{bmatrix} 0 & 0 & 0 & -1 \\ 0 & 0 & -1 & 0 \\ 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix} \\ &= a_0 I_4 + a_1 M_1 + a_2 M_2 + a_3 M_3. \end{aligned} \tag{4}$$

I_4 is the identity, and M_1, M_2, and M_3 have the properties:

$$\begin{aligned} M_i^T &= -M_i, \\ M_i^T M_i &= I, \\ M_i M_i &= -I, \quad \text{where } M_i^T \text{ is transpose of } M_i,\ i = 1, 2, 3. \end{aligned} \tag{5}$$

Table 1: Imaginary units multiplication.

	i_1	i_2	i_3
i_1	-1	i_3	$-i_2$
i_2	$-i_3$	-1	i_1
i_3	i_2	$-i_1$	-1

A square matrix whose transpose equals its negative is said to be skew-symmetric (or antimetric) [8], M_1, M_2, and M_3 are 4×4 skew-symmetric matrices.

From the basic properties of transposes [9], we know that if M is skew-symmetric matrix, then the transpose M^T is skew-symmetric too, so the matrix representation of a quaternion number is not unique.

The transpose matrix of A_1 gives another representation of a quaternion number a:

$$\begin{aligned} B_1 = A_1^T &= \begin{bmatrix} a_0 & a_1 & a_2 & a_3 \\ -a_1 & a_0 & a_3 & -a_2 \\ -a_2 & -a_3 & a_0 & a_1 \\ -a_3 & a_2 & -a_1 & a_0 \end{bmatrix} \\ &= a_0 \begin{bmatrix} 1&0&0&0\\0&1&0&0\\0&0&1&0\\0&0&0&1 \end{bmatrix} + a_1 \begin{bmatrix} 0&1&0&0\\-1&0&0&0\\0&0&0&-1\\0&0&1&0 \end{bmatrix} \\ &\quad + a_2 \begin{bmatrix} 0&0&1&0\\0&0&0&-1\\-1&0&0&0\\0&1&0&0 \end{bmatrix} + a_3 \begin{bmatrix} 0&0&0&1\\0&0&1&0\\0&-1&0&0\\-1&0&0&0 \end{bmatrix} \\ &= a_0 I_4 + a_1 N_1 + a_2 N_2 + a_3 N_3. \end{aligned} \tag{6}$$

N_1, N_2, and N_3 are 4×4 skew-symmetric matrices.

The purpose of this paper is to eliminate the confusion that existing about the total number of real matrices that can represent a quaternion number.

We investigate the questions:

(1) How many distinct matrices in the form of A_1 can give representation for a quaternion number?

(2) How to determine these matrices?

We will consider two sets of matrices:

Left matrix representation and its transpose

Right matrix representation and its transpose

Farebrother et al. [10] take one of these two sets and talk about 48 distinct matrices representation for a quaternion number.

2. Discussion and Results

Let us split the matrix A_1 into four 2×2 matrices; of course each matrix has an element takes a different sign from the others:

$$A_1 = \begin{bmatrix} \varphi_1(a) & -\varphi_2(b) \\ \varphi_2(b) & \varphi_1(a) \end{bmatrix}, \tag{7}$$

where $\varphi_1(a) = \begin{bmatrix} a_0 & -a_1 \\ a_1 & a_0 \end{bmatrix}$ and $\varphi_2(b) = \begin{bmatrix} a_2 & a_3 \\ a_3 & -a_2 \end{bmatrix}$.

φ_1 has 6 forms:

$$\begin{aligned} &\begin{bmatrix} a_0 & -a_1 \\ a_1 & a_0 \end{bmatrix}, \begin{bmatrix} a_0 & a_1 \\ -a_1 & a_0 \end{bmatrix}, \begin{bmatrix} a_0 & -a_2 \\ a_2 & a_0 \end{bmatrix}, \begin{bmatrix} a_0 & a_2 \\ -a_2 & a_0 \end{bmatrix}, \\ &\begin{bmatrix} a_0 & -a_3 \\ a_3 & a_0 \end{bmatrix}, \begin{bmatrix} a_0 & a_3 \\ -a_3 & a_0 \end{bmatrix}, \end{aligned} \tag{8}$$

whereas φ_2 (two elements remaining, say a_2 and a_3) has 16 forms:

$$\begin{aligned} &\begin{bmatrix} a_2 & a_3 \\ a_3 & -a_2 \end{bmatrix}, \begin{bmatrix} -a_2 & a_3 \\ a_3 & a_2 \end{bmatrix}, \begin{bmatrix} a_2 & -a_3 \\ a_3 & a_2 \end{bmatrix}, \begin{bmatrix} a_2 & a_3 \\ -a_3 & a_2 \end{bmatrix}, \\ &\begin{bmatrix} a_2 & -a_3 \\ -a_3 & -a_2 \end{bmatrix}, \begin{bmatrix} -a_2 & -a_3 \\ -a_3 & a_2 \end{bmatrix}, \begin{bmatrix} -a_2 & a_3 \\ -a_3 & -a_2 \end{bmatrix}, \begin{bmatrix} -a_2 & -a_3 \\ a_3 & -a_2 \end{bmatrix}, \\ &\begin{bmatrix} a_3 & a_2 \\ a_2 & -a_3 \end{bmatrix}, \begin{bmatrix} -a_3 & a_2 \\ a_2 & a_3 \end{bmatrix}, \begin{bmatrix} a_3 & -a_2 \\ a_2 & a_3 \end{bmatrix}, \begin{bmatrix} a_3 & a_2 \\ -a_2 & a_3 \end{bmatrix}, \\ &\begin{bmatrix} a_3 & -a_2 \\ -a_2 & -a_3 \end{bmatrix}, \begin{bmatrix} -a_3 & -a_2 \\ -a_2 & a_3 \end{bmatrix}, \begin{bmatrix} -a_3 & a_2 \\ -a_2 & -a_3 \end{bmatrix}, \begin{bmatrix} -a_3 & -a_2 \\ a_2 & -a_3 \end{bmatrix}. \end{aligned} \tag{9}$$

Therefore, the total number of all possible matrices is $(6)(16) = 96$.

To determine these matrices, we do some operations on both columns and rows of the matrix A_1(Tables 2–4 and Algorithms 1–6). We will find

Table 2: One switching operation.

Operation	Model	Number of all possible matrices
(1) One switching: selecting two columns to switch them together		$4!/2!2! = 6$

Table 3: Two switching operations.

Operation	Model	Number of all possible matrices
(2) Two switching:		
(i) Selecting two columns to switch them, and switching the other two columns together		$(4!/2!2!)(2!/0!2!)/2 = 3$
(ii) Selecting three columns then selecting one of them to switch with the others		$(4!/1!3!)(2) = 8$

Table 4: Three switching operations.

Operation	Model	Number of all possible matrices
(3) Three switching: selecting a column to switch with a second column, then switching this second column with a third column, and switching this third column with the remaining column		6

```
#Python code for one column (row) switching
def OneSwitching (A, n, m):
    for i in range (4):
        R = A[n][i]
        A[n][i] = A[m][i]
        A[m][i] = R
    for j in range (4):
        C = A[j][n]
        A[j][n] = A[j][m]
        A[j][m] = C
m = 1 # Replacing m by 2 and another time by 3 gives all 6 matrices
for n in range (m):
    A = [['a0', '−a1', '−a2', '−a3'], ['a1', 'a0', '−a3', 'a2'], ['a2', 'a3', 'a0', '−a1'], ['a3', '−a2', 'a1', 'a0']]
    print ("The Matrix = n")
    OneSwitching (A, n, m)
    for i in range (4):
        for j in range (4):
            print (A[i][j], end = " ")
        print ("\n")
```

Algorithm 1: One switching 1.

```
#Python code for two columns (rows) switching
def TwoSwitching1 (A, n, m):
    for i in range (4):
        R1 = A[0][i]
        A[0][i] = A[2 * m − n − 1][i]
        A[2 * m − n − 1][i] = R1
        R2 = A[2 * k + m − 1][i]
        A[2 * k + m − 1][i] = A[2 * k + m + n][i]
        A[2 * k + m + n][i] = R2
    for j in range (4):
        C1 = A[j][0]
        A[j][0] = A[j][2 * m − n − 1]
        A[j][2 * m − n − 1] = C1
        C2 = A[j][2 * k + m − 1]
        A[j][2 * k + m − 1] = A[j][2 * k + m + n]
```

Algorithm 2: Continued.

```
        A[j][2 * k + m + n] = C2
m = 2
k = 0 # another time put m = 1 and k = 1
for n in range (m):
    A = [['a0', '−a1', '−a2', '−a3'], ['a1', 'a0', '−a3', 'a2'], ['a2', 'a3', 'a0', '−a1'], ['a3', '−a2', 'a1', 'a0']]
    print ("The Matrix = n")
    TwoSwitching1 (A, n, m)
    for i in range (4):
        for j in range (4):
            print (A[i][j], end = " ")
        print ("\n")
```

Algorithm 2: Two switching operations 1.

```
#Python code for two columns (rows) switching
def TwoSwitching2 (A, n, m):
    for i in range (4):
        R = A[0][i]
        A[0][i] = A[2 * k + 1][i]
        A[2 * k + 1][i] = A[n + m − k][i]
        A[n + m − k][i] = R
    for j in range (4):
        C = A[j][0]
        A[j][0] = A[j][2 * k + 1]
        A[j][2 * k + 1] = A[j][n + m − k]
        A[j][n + m − k] = C
m = 2
k = 0 # another time put k = 1
for n in range (m):
    A = [['a0', '−a1', '−a2', '−a3'], ['a1', 'a0', '−a3', 'a2'], ['a2', 'a3', 'a0', '−a1'], ['a3', '−a2', 'a1', 'a0']]
    print ("The Matrix = n")
    TwoSwitching2 (A, n, m)
    for i in range (4):
        for j in range (4):
            print (A[i][j], end = " ")
        print ("\n")
```

Algorithm 3: Two switching 2.

```
#Python code for two columns (rows) switching
def TwoSwitching3 (A, n, m):
    for i in range (4):
        R = A[n][i]
        A[n][i] = A[k * n + m][i]
        A[k * n + m][i] = A[k * (n − 2) + m + 1][i]
        A[k * (n − 2) + m + 1][i] = R
    for j in range (4):
        C = A[j][n]
        A[j][n] = A[j][k * n + m]
        A[j][k * n + m] = A[j][k * (n − 2) + m + 1]
        A[j][k * (n − 2) + m + 1] = C
m = 2
k = 0 # another time put k = 1
for n in range (m):
    A = [['a0', '−a1', '−a2', '−a3'], ['a1', 'a0', '−a3', 'a2'], ['a2', 'a3', 'a0', '−a1'], ['a3', '−a2', 'a1', 'a0']]
    print ("The Matrix =n")
```

Algorithm 4: Continued.

```
TwoSwitching3 (A, n, m)
for i in range (4):
    for j in range (4):
        print (A[i][j], end = " ")
    print ("\n")
```

Algorithm 4: Two switching 3.

```
#Python code for three columns (rows) switching
def ThreeSwitching1 (A, n, m):
    for i in range (4):
        R = A[0][i]
        A[0][i] = A[2 * k + 1][i]
        A[2 * k + 1][i] = A[n + m − k][i]
        A[n + m − k][i] = A[2 * m − n − k − 1][i]
        A[2 * m − n − k − 1][i] = R
    for j in range (4):
        C = A[j][0]
        A[j][0] = A[j][2 * k + 1]
        A[j][2 * k + 1] = A[j][n + m − k]
        A[j][n + m − k] = A[j][2 * m − n − k − 1]
        A[j][2 * m − n − k − 1] = C
m = 2
k = 0 # another time k = 1
for n in range (m):
    A = [['a0', '−a1', '−a2', '−a3'], ['a1', 'a0', '−a3', 'a2'], ['a2', 'a3', 'a0', '−a1'], ['a3', '−a2', 'a1', 'a0']]
    print ("The Matrix =n")
    ThreeSwitching1 (A, n, m)
    for i in range (4):
        for j in range(4):
            print (A[i][j], end = " ")
        print ("\n")
```

Algorithm 5: Three switching 1.

```
#Python code for three columns (rows) switching
def ThreeSwitching2 (A, n, m):
    for i in range (4):
        R = A[n][i]
        A[n][i] = A[m − 2 * n][i]
        A[m − 2 * n][i] = A[n + m − 1][i]
        A[n + m − 1][i] = A[m + 1][i]
        A[m + 1][i] = R
    for j in range (4):
        C = A[j][n]
        A[j][n] = A[j][m − 2 * n]
        A[j][m − 2 * n] = A[j][n + m − 1]
        A[j][n + m − 1] = A[j][m + 1]
        A[j][m + 1] = C
m = 2
for n in range (m):
    A = [['a0', '−a1', '−a2', '−a3'], ['a1', 'a0', '−a3', 'a2'], ['a2', 'a3', 'a0', '−a1'], ['a3', '−a2', 'a1', 'a0']]
    print ("The Matrix =n")
    ThreeSwitching2 (A, n, m)
    for i in range (4):
        for j in range    (4):
            print (A[i][j], end = " ")
        print ("\n")
```

Algorithm 6: Three switching 2.

$$
\begin{aligned}
&C_1 \longleftrightarrow C_2, R_1 \longleftrightarrow R_2,\\
&A_2 = \begin{bmatrix} a_0 & a_1 & -a_3 & a_2 \\ -a_1 & a_0 & -a_2 & -a_3 \\ a_3 & a_2 & a_0 & -a_1 \\ -a_2 & a_3 & a_1 & a_0 \end{bmatrix},\\
&C_1 \longleftrightarrow C_3, R_1 \longleftrightarrow R_3,\\
&A_3 = \begin{bmatrix} a_0 & a_3 & a_2 & -a_1 \\ -a_3 & a_0 & a_1 & a_2 \\ -a_2 & -a_1 & a_0 & -a_3 \\ a_1 & -a_2 & a_3 & a_0 \end{bmatrix},\\
&C_1 \longleftrightarrow C_4, R_1 \longleftrightarrow R_4,\\
&A_4 = \begin{bmatrix} a_0 & -a_2 & a_1 & a_3 \\ a_2 & a_0 & -a_3 & a_1 \\ -a_1 & a_3 & a_0 & a_2 \\ -a_3 & -a_1 & -a_2 & a_0 \end{bmatrix},\\
&C_2 \longleftrightarrow C_3, R_2 \longleftrightarrow R_3,\\
&A_5 = \begin{bmatrix} a_0 & -a_2 & -a_1 & -a_3 \\ a_2 & a_0 & a_3 & -a_1 \\ a_1 & -a_3 & a_0 & a_2 \\ a_3 & a_1 & -a_2 & a_0 \end{bmatrix},\\
&C_2 \longleftrightarrow C_4, R_2 \longleftrightarrow R_4,\\
&A_6 = \begin{bmatrix} a_0 & -a_3 & -a_2 & -a_1 \\ a_3 & a_0 & a_1 & -a_2 \\ a_2 & -a_1 & a_0 & a_3 \\ a_1 & a_2 & -a_3 & a_0 \end{bmatrix},\\
&C_3 \longleftrightarrow C_4, R_3 \longleftrightarrow R_4,\\
&A_7 = \begin{bmatrix} a_0 & -a_1 & -a_3 & -a_2 \\ a_1 & a_0 & a_2 & -a_3 \\ a_3 & -a_2 & a_0 & a_1 \\ a_2 & a_3 & -a_1 & a_0 \end{bmatrix},
\end{aligned}
\tag{10}
$$

$$
\begin{aligned}
&C_1 \longleftrightarrow C_2, C_3 \longleftrightarrow C_4,\\
&R_1 \longleftrightarrow R_2, R_3 \longleftrightarrow R_4,\\
&A_8 = \begin{bmatrix} a_0 & a_1 & a_2 & -a_3 \\ -a_1 & a_0 & -a_3 & -a_2 \\ -a_2 & a_3 & a_0 & a_1 \\ a_3 & a_2 & -a_1 & a_0 \end{bmatrix},\\
&C_1 \longleftrightarrow C_3, C_2 \longleftrightarrow C_4,\\
&R_1 \longleftrightarrow R_3, R_2 \longleftrightarrow R_4,\\
&A_9 = \begin{bmatrix} a_0 & -a_1 & a_2 & a_3 \\ a_1 & a_0 & a_3 & -a_2 \\ -a_2 & -a_3 & a_0 & -a_1 \\ -a_3 & a_2 & a_1 & a_0 \end{bmatrix},\\
&C_1 \longleftrightarrow C_4, C_2 \longleftrightarrow C_3,\\
&R_1 \longleftrightarrow R_4, R_2 \longleftrightarrow R_3,\\
&A_{10} = \begin{bmatrix} a_0 & a_1 & -a_2 & a_3 \\ -a_1 & a_0 & a_3 & a_2 \\ a_2 & -a_3 & a_0 & a_1 \\ -a_3 & -a_2 & -a_1 & a_0 \end{bmatrix},
\end{aligned}
\tag{11}
$$

$$
\begin{aligned}
&C_1 \longleftrightarrow C_2 \longleftrightarrow C_3,\\
&R_1 \longleftrightarrow R_2 \longleftrightarrow R_3,\\
&A_{11} = \begin{bmatrix} a_0 & a_2 & a_3 & -a_1 \\ -a_2 & a_0 & -a_1 & -a_3 \\ -a_3 & a_1 & a_0 & a_2 \\ a_1 & a_3 & -a_2 & a_0 \end{bmatrix},\\
&C_1 \longleftrightarrow C_2 \longleftrightarrow C_4,\\
&R_1 \longleftrightarrow R_2 \longleftrightarrow R_4,\\
&A_{12} = \begin{bmatrix} a_0 & a_3 & a_1 & -a_2 \\ -a_3 & a_0 & -a_2 & -a_1 \\ -a_1 & a_2 & a_0 & a_3 \\ a_2 & a_1 & -a_3 & a_0 \end{bmatrix},\\
&C_1 \longleftrightarrow C_4 \longleftrightarrow C_2,\\
&R_1 \longleftrightarrow R_4 \longleftrightarrow R_2,\\
&A_{13} = \begin{bmatrix} a_0 & a_2 & -a_3 & a_1 \\ -a_2 & a_0 & a_1 & a_3 \\ a_3 & -a_1 & a_0 & a_2 \\ -a_1 & -a_3 & -a_2 & a_0 \end{bmatrix},\\
&C_1 \longleftrightarrow C_4 \longleftrightarrow C_3,\\
&R_1 \longleftrightarrow R_4 \longleftrightarrow R_3,\\
&A_{14} = \begin{bmatrix} a_0 & a_3 & -a_1 & a_2 \\ -a_3 & a_0 & a_2 & a_1 \\ a_1 & -a_2 & a_0 & a_3 \\ -a_2 & -a_1 & -a_3 & a_0 \end{bmatrix},\\
&C_1 \longleftrightarrow C_3 \longleftrightarrow C_4,\\
&R_1 \longleftrightarrow R_3 \longleftrightarrow R_4,\\
&A_{15} = \begin{bmatrix} a_0 & -a_2 & a_3 & a_1 \\ a_2 & a_0 & a_1 & -a_3 \\ -a_3 & -a_1 & a_0 & -a_2 \\ -a_1 & a_3 & a_2 & a_0 \end{bmatrix},\\
&C_2 \longleftrightarrow C_3 \longleftrightarrow C_4,\\
&R_2 \longleftrightarrow R_3 \longleftrightarrow R_4,\\
&A_{16} = \begin{bmatrix} a_0 & -a_3 & -a_1 & -a_2 \\ a_3 & a_0 & -a_2 & a_1 \\ a_1 & a_2 & a_0 & -a_3 \\ a_2 & -a_1 & a_3 & a_0 \end{bmatrix},\\
&C_1 \longleftrightarrow C_3 \longleftrightarrow C_2,\\
&R_1 \longleftrightarrow R_3 \longleftrightarrow R_2,\\
&A_{17} = \begin{bmatrix} a_0 & -a_3 & a_1 & a_2 \\ a_3 & a_0 & a_2 & -a_1 \\ -a_1 & -a_2 & a_0 & -a_3 \\ -a_2 & a_1 & a_3 & a_0 \end{bmatrix},\\
&C_2 \longleftrightarrow C_4 \longleftrightarrow C_3,\\
&R_2 \longleftrightarrow R_4 \longleftrightarrow R_3,\\
&A_{18} = \begin{bmatrix} a_0 & -a_2 & -a_3 & a_1 \\ a_2 & a_0 & -a_1 & a_3 \\ a_3 & a_1 & a_0 & -a_2 \\ a_1 & -a_3 & a_2 & a_0 \end{bmatrix},
\end{aligned}
\tag{12}
$$

$$
\begin{aligned}
&C_1 \longleftrightarrow C_2 \longleftrightarrow C_3 \longleftrightarrow C_4,\\
&R_1 \longleftrightarrow R_2 \longleftrightarrow R_3 \longleftrightarrow R_4,\\
&A_{19} = \begin{bmatrix} a_0 & -a_3 & a_2 & a_1 \\ a_3 & a_0 & -a_1 & a_2 \\ -a_2 & a_1 & a_0 & a_3 \\ -a_1 & -a_2 & -a_3 & a_0 \end{bmatrix},\\
&C_1 \longleftrightarrow C_2 \longleftrightarrow C_4 \longleftrightarrow C_3,\\
&R_1 \longleftrightarrow R_2 \longleftrightarrow R_4 \longleftrightarrow R_3,\\
&A_{20} = \begin{bmatrix} a_0 & a_2 & a_1 & -a_3 \\ -a_2 & a_0 & a_3 & a_1 \\ -a_1 & -a_3 & a_0 & -a_2 \\ a_3 & -a_1 & a_2 & a_0 \end{bmatrix},\\
&C_1 \longleftrightarrow C_4 \longleftrightarrow C_2 \longleftrightarrow C_3,\\
&R_1 \longleftrightarrow R_4 \longleftrightarrow R_2 \longleftrightarrow R_3,\\
&A_{21} = \begin{bmatrix} a_0 & a_3 & -a_2 & a_1 \\ -a_3 & a_0 & -a_1 & -a_2 \\ a_2 & a_1 & a_0 & -a_3 \\ -a_1 & a_2 & a_3 & a_0 \end{bmatrix},\\
&C_1 \longleftrightarrow C_4 \longleftrightarrow C_3 \longleftrightarrow C_2,\\
&R_1 \longleftrightarrow R_4 \longleftrightarrow R_3 \longleftrightarrow R_2,\\
&A_{22} = \begin{bmatrix} a_0 & a_1 & a_3 & -a_2 \\ -a_1 & a_0 & a_2 & a_3 \\ -a_3 & -a_2 & a_0 & -a_1 \\ a_2 & -a_3 & a_1 & a_0 \end{bmatrix},\\
&C_1 \longleftrightarrow C_3 \longleftrightarrow C_2 \longleftrightarrow C_4,\\
&R_1 \longleftrightarrow R_3 \longleftrightarrow R_2 \longleftrightarrow R_4,\\
&A_{23} = \begin{bmatrix} a_0 & -a_1 & a_3 & a_2 \\ a_1 & a_0 & -a_2 & a_3 \\ -a_3 & a_2 & a_0 & a_1 \\ -a_2 & -a_3 & -a_1 & a_0 \end{bmatrix},\\
&C_2 \longleftrightarrow C_1 \longleftrightarrow C_3 \longleftrightarrow C_4,\\
&R_2 \longleftrightarrow R_1 \longleftrightarrow R_3 \longleftrightarrow R_4,\\
&A_{24} = \begin{bmatrix} a_0 & a_2 & -a_1 & a_3 \\ -a_2 & a_0 & -a_3 & -a_1 \\ a_1 & a_3 & a_0 & -a_2 \\ -a_3 & a_1 & a_2 & a_0 \end{bmatrix}.
\end{aligned}
\tag{13}
$$

We obtained 23 distinct matrices from the matrix A_1, and by similar way, we can obtain another 23 matrices from the matrix B_1.

Also, we can get another 46 matrices from C_1 and D_1.

We stablish C_1 by multiplying (2) from the left by i_1, i_2, and i_3, respectively,

$$
\begin{aligned}
i_1 \cdot a &= a_0 i_1 - a_1 + a_2 i_3 - a_3 i_2,\\
i_2 \cdot a &= a_0 i_2 - a_1 i_3 - a_2 + a_3 i_1,\\
i_3 \cdot a &= a_0 i_3 + a_1 i_2 - a_2 i_1 - a_3.
\end{aligned}
\tag{14}
$$

So,

$$
\begin{aligned}
C_1 &= \begin{bmatrix} a_0 & -a_1 & -a_2 & -a_3 \\ a_1 & a_0 & a_3 & -a_2 \\ a_2 & -a_3 & a_0 & a_1 \\ a_3 & a_2 & -a_1 & a_0 \end{bmatrix} = a_0 \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} + a_1 \begin{bmatrix} 0 & -1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 \\ 0 & 0 & 1 & 0 \end{bmatrix}\\
&\quad + a_2 \begin{bmatrix} 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & -1 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} + a_3 \begin{bmatrix} 0 & 0 & 0 & -1 \\ 0 & 0 & 1 & 0 \\ 0 & -1 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix} = a_0 I_4 + a_1 S_1 + a_2 S_2 + a_3 S_3.
\end{aligned}
\tag{15}
$$

$S_1, S_2,$ and S_3 are 4×4 skew-symmetric matrices, whereas D_1 is the transpose of C_1:

$$D_1 = C_1^T = \begin{bmatrix} a_0 & a_1 & a_2 & a_3 \\ -a_1 & a_0 & -a_3 & a_2 \\ -a_2 & a_3 & a_0 & -a_1 \\ -a_3 & -a_2 & a_1 & a_0 \end{bmatrix} = a_0 \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} + a_1 \begin{bmatrix} 0 & 1 & 0 & 0 \\ -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 \\ 0 & 0 & 1 & 0 \end{bmatrix} + a_2 \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ -1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \end{bmatrix} + a_3 \begin{bmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & -1 & 0 \\ 0 & 1 & 0 & 0 \\ -1 & 0 & 0 & 0 \end{bmatrix} = a_0 I_4 + a_1 T_1 + a_2 T_2 + a_3 T_3. \tag{16}$$

$T_1, T_2,$ and T_3 are 4×4 skew-symmetric matrices. We put these 96 matrices into two sets:

Left matrices representation set which consists of A_{i} and D_{i}

Right matrices representation set which consists of B_{i} and C_{i} where $(1 \le i \le 24)$

In solving quaternions problems, we have to pay attention to the difference between the two sets, for example, if we want to find the product ab, where $a = a_0 + a_1 i_1 + a_2 i_2 + a_3 i_3$ and $b = b_0 + b_1 i_1 + b_2 i_2 + b_3 i_3$.

(1) By using left matrices representation:

If we take A_{24} (for example),

$$\begin{bmatrix} a_0 & a_2 & -a_1 & a_3 \\ -a_2 & a_0 & -a_3 & -a_1 \\ a_1 & a_3 & a_0 & -a_2 \\ -a_3 & a_1 & a_2 & a_0 \end{bmatrix} \begin{bmatrix} b_0 \\ -b_2 \\ b_1 \\ -b_3 \end{bmatrix} = \begin{bmatrix} a_0b_0 - a_2b_2 - a_1b_1 - a_3b_3 \\ -a_2b_0 - a_0b_2 - a_3b_1 + a_1b_3 \\ a_1b_0 - a_3b_2 + a_0b_1 + a_2b_3 \\ -a_3b_0 - a_1b_2 + a_2b_1 - a_0b_3 \end{bmatrix}. \tag{17}$$

Thus,

$$\begin{aligned} ab = {} & (a_0b_0 - a_2b_2 - a_1b_1 - a_3b_3) + (a_1b_0 - a_3b_2 + a_0b_1 + a_2b_3)i_1 \\ & - (-a_2b_0 - a_0b_2 - a_3b_1 + a_1b_3)i_2 - (-a_3b_0 - a_1b_2 + a_2b_1 - a_0b_3)i_3. \end{aligned} \tag{18}$$

If we take D_1,

$$\begin{bmatrix} a_0 & a_1 & a_2 & a_3 \\ -a_1 & a_0 & -a_3 & a_2 \\ -a_2 & a_3 & a_0 & -a_1 \\ -a_3 & -a_2 & a_1 & a_0 \end{bmatrix} \begin{bmatrix} b_0 \\ -b_1 \\ -b_2 \\ -b_3 \end{bmatrix} = \begin{bmatrix} a_0b_0 - a_1b_1 - a_2b_2 - a_3b_3 \\ -a_1b_0 - a_0b_1 + a_3b_2 - a_2b_3 \\ -a_2b_0 - a_3b_1 - a_0b_2 + a_1b_3 \\ -a_3b_0 + a_2b_1 - a_1b_2 - a_0b_3 \end{bmatrix}. \tag{19}$$

Thus,

$$\begin{aligned} ab = {} & (a_0b_0 - a_1b_1 - a_2b_2 - a_3b_3) - (-a_1b_0 - a_0b_1 + a_3b_2 - a_2b_3)i_1 \\ & - (-a_2b_0 - a_3b_1 - a_0b_2 + a_1b_3)i_2 - (-a_3b_0 + a_2b_1 - a_1b_2 - a_0b_3)i_3. \end{aligned} \tag{20}$$

(2) By using right matrices representation:

If we take B_1,

$$\begin{bmatrix} b_0 & b_1 & b_2 & b_3 \\ -b_1 & b_0 & b_3 & -b_2 \\ -b_2 & -b_3 & b_0 & b_1 \\ -b_3 & b_2 & -b_1 & b_0 \end{bmatrix} \begin{bmatrix} a_0 \\ -a_1 \\ -a_2 \\ -a_3 \end{bmatrix} = \begin{bmatrix} a_0b_0 - a_1b_1 - a_2b_2 - a_3b_3 \\ -a_0b_1 - a_1b_0 - a_2b_3 + a_3b_2 \\ -a_0b_2 + a_1b_3 - a_2b_0 - a_3b_1 \\ -a_0b_3 - a_1b_2 + a_2b_1 - a_3b_0 \end{bmatrix}. \tag{21}$$

Thus,

$$\begin{aligned} ab = {} & (a_0b_0 - a_1b_1 - a_2b_2 - a_3b_3) - (-a_0b_1 - a_1b_0 - a_2b_3 + a_3b_2)i_1 \\ & - (-a_0b_2 + a_1b_3 - a_2b_0 - a_3b_1)i_2 - (-a_0b_3 - a_1b_2 + a_2b_1 - a_3b_0)i_3. \end{aligned} \tag{22}$$

If we take C_1,

$$\begin{bmatrix} b_0 & -b_1 & -b_2 & -b_3 \\ b_1 & b_0 & b_3 & -b_2 \\ b_2 & -b_3 & b_0 & b_1 \\ b_3 & b_2 & -b_1 & b_0 \end{bmatrix} \begin{bmatrix} a_0 \\ a_1 \\ a_2 \\ a_3 \end{bmatrix} = \begin{bmatrix} a_0b_0 - a_1b_1 - a_2b_2 - a_3b_3 \\ a_0b_1 + a_1b_0 + a_2b_3 - a_3b_2 \\ a_0b_2 - a_1b_3 + a_2b_0 + a_3b_1 \\ a_0b_3 + a_1b_2 - a_2b_1 + a_3b_0 \end{bmatrix}. \tag{23}$$

Thus,

$$\begin{aligned} ab = {} & (a_0b_0 - a_2b_2 - a_1b_1 - a_3b_3) + (a_0b_1 + a_1b_0 + a_2b_3 - a_3b_2)i_1 \\ & + (a_0b_2 - a_1b_3 + a_2b_0 + a_3b_1)i_2 + (a_0b_3 + a_1b_2 - a_2b_1 + a_3b_0)i_3. \end{aligned} \tag{24}$$

Is there any advantage in choosing specific matrix representation of a quaternion number in solving a problem? This is the third question, and it will be left for future work.

3. Conclusion

Each one of the four real matrices, A_1 (left matrix representation), B_1 (transpose matrix of A_1), C_1 (right matrix representation), and D_1 (transpose matrix of C_1), gives 23 distinct matrices, so precisely, there are 96 real matrices that represent a quaternion number where each one of them consists of three 4×4 skew-symmetric matrices plus the identity matrix.

Data Availability

No data were used to support the findings of the study.

Conflicts of Interest

The author declares no conflicts of interest.

References

[1] T.-Y. Lam, *Hamilton's Quaternions*, Handbook of Algebra, Amsterdam, The Netherlands, 2003.
[2] D. W. Lewis, "Quaternion algebras and the algebraic legacy of Hamilton's quaternions," *Irish Mathematical Society*, vol. 57, pp. 41–64, 2006.
[3] W. R. Hamilton, "On quaternions or a new system of imaginaries in algebra, philosophical magazine," 2000.
[4] R. Goldman, "Understanding quaternions," *Graphical Models*, vol. 73, no. 2, p. 21, 2011.
[5] J. Pujol, "On Hamilton's nearly-forgotten early work on the relation between rotations and quaternions and on the composition of rotations," *American Mathematical Monthly*, vol. 121, 2014.

[6] A. van Weerden and S. Wepster, "A most gossiped about genius: Sir William Rowan Hamilton," *BSHM Bulletin: Journal of the British Society for the History of Mathematics*, vol. 33, no. 1, pp. 2–20, 2018.
[7] J. H. Conway and D. A. Smith, "On quaternions and octonions," *AMC*, vol. 10, p. 12, 2003.
[8] R. A. Reyment, K. G. Jöreskog, and L. F. Marcus, *Applied Factor Analysis in the Natural Sciences*, Cambridge University Press, London, UK, 1996.
[9] T. A. Whitelaw, *Introduction to Linear Algebra*, CRC Press, London, UK, 2nd edition, 1991.
[10] R. W. Farebrother, J. Groß, and S.-O. Troschke, "Matrix representation of quaternions," *Linear Algebra and Its Applications*, vol. 362, pp. 251–255, 2003.