IOSR Journal of Mathematics (IOSR-JM)
e-ISSN: 2278-5728, p-ISSN: 2319-765X. Volume 14, Issue 3 Ver. III (May - June 2018), PP 38-43
www.iosrjournals.org

Real Representation and Its Applications in Split Quaternionic Mechanics

Minghui Wang[1*], Xiaowen Cui[1], Xuehan Cheng[2]
[1]*(Department of Mathematics, Qingdao University of Science and Technology, Qingdao 266061,PR China)*
[2]*(Department of Mathematics, Ludong University, Yantai, 264025, PR China)*
Corresponding Author: Minghui Wang

Abstract: *A real representation of the split quaternion matrix is defined and its properties are studied. The differences between three representations are discussed. Then the split quaternionic least squares problem is studied in detail from norm, definition to computation, and an algebraic method is obtained for finding solutions of the problem in split quaternionic mechanics. Numerical examples show that our method is effective and better than that of the paper[Z. Zhang, Z. Jiang, T. Jiang, Algebraic methods for least squares problem in split quaternionic mechanics, Applied Mathematics and Computation, 269 (2015), 618-625.]*

Keywords: *split quaternion matrix; real presentation; split quaternionic least squares*

Date of Submission: 05-06-2018 Date of acceptance: 24-06-2018

I. INTRODUCTION

The introduction of the paper should explain the nature of the problem, previous work, purpose, and the contribution of the paper. The contents of each section may be provided to understand easily about the paper. (10) The quaternion invented by William Rowan Hamilton(1805-1865) has been widely used in quaternionic quantum mechanics and many other fileds. In 1849, James Cockle found the split quaternion, which has the following form:

$$q = q_0 + q_1 i + q_2 j + q_3 k, i^2 = -1, j^2 = k^2 = 1, ijk = 1,$$

where q_0, q_1, q_2, q_3 are real and $ij = -ji = k, jk = -kj = -i, ki = -ik = j$. The set of all split quaternions is a ring, denoted by SQ. This ring is an associative, noncommutative four-dimensional Clifford algebra and has zero divisors, nilpotent elements and nontrivial idempotents[1]. SQ is different from the quaternion ring and has more complicated algebraic structure. For details, see [1] and the references therein.

In complexified classical and non-Hermitian quantum mechanics, there are surprising relations between quaternionic and split quaternionic mechanics [2]. In the literature over the past decade, the complexified mechanical systems with real energies are studied extensively, which can alternatively be viewed as certain split quaternionic extensions of the underlying real mechanical systems [3]. This result leads to the possibility of employing algebraic techniques of quaternions and split quaternions to deal with some challenging open issues in complexified classical quantum mechanics.

Let R and $\mathbf{SQ} = \mathbf{R} \oplus \mathbf{R}i \oplus \mathbf{R}j \oplus \mathbf{R}k$ denote the real number field and the split quaternion ring, respectively. For $a = a_0 + a_1 i + a_2 j + a_3 k, b = b_0 + b_1 i + b_2 j + b_3 k \in \mathbf{SQ}$, the conjugate of a is defined as $\bar{a} = a_0 - a_1 i - a_2 j - a_3 k$, then $a\bar{a} = a_0^2 + a_1^2 - a_2^2 - a_3^2$. The module $|a|$ of a split quaternion a is defined as $|a| = \sqrt{|a\bar{a}|} = \sqrt{|a_0^2 + a_1^2 - a_2^2 - a_3^2|}$. a is said to be a unit split quaternion if its norm is 1.and their multiplication is defined as

$$\begin{aligned} ab = &(a_0 b_0 - a_1 b_1 + a_2 b_2 + a_3 b_3) + (a_0 b_1 + a_1 b_0 - a_2 b_3 + a_3 b_2) i \\ &+ (a_0 b_2 + a_2 b_0 - a_1 b_3 + a_3 b_1) j + (a_0 b_3 + a_3 b_0 + a_1 b_2 - a_2 b_1) k. \end{aligned}$$

For any quaternion matrix A, A^T and A^H denote the transpose, and conjugate transpose of A, respectively. $A(i:j, k:l)$ represents the submatrix of A containing the intersection of rows i to j and columns k to l. I_n denotes the unit matrix of order n.

In this paper, we will define a real presentation of the split quaternion matrix and study its properties. Also we will discuss the differences between three representations: the real presentation and the complex presentation of [4] and our real presentation. As the first application of our real presentation, then we will give an alternative of the split quaternion norm, which enable us to define and study the split quaternionic least squares(SQLS) problem. An algebraic method is obtained for finding solutions of the SQLS problem in split quaternionic mechanics. As the second application, we will discuss the inverse and its computation of the split quaternion matrix. At last, numerical examples show that, for some special problems, our method is effective and better than that of the paper[4].

This paper is organized as follows. In Section 2, the real representation will be defined and discussed. In Section 3, the SQLS problem and the inverse of the split quaternion matrix will be studied. In Section 4, two numerical experiments will be provided to demonstrate the efficiency of our algorithm. Finally, some concluding remarks will given in Section 5.

II. REAL REPRESENTATION

In this section, we will give the definition of the real presentation and discuss its properties.
Let $A_l \in \mathbf{R}^{m\times n} (l = 0,1,2,3).$ The real representation matrix is defined as follows.

$$A^R @ \begin{pmatrix} A_0 + A_2 & -A_1 + A_3 \\ A_1 + A_3 & A_0 - A_2 \end{pmatrix} \in \mathbf{R}^{2m\times 2n}. \qquad (2.1)$$

The real matrix A^R in (2.1) is uniquely determined by the split quaternion matrix $A = A_0 + A_1 i + A_2 j + A_3 k \in \mathbf{SQ}^{m\times n}$, and it is said to be a real representation matrix of the split quaternion matrix A .

By simple computation, we can obtain the following properties.

Theorem 2.1 Let $A, B \in \mathbf{SQ}^{m\times n}, C \in \mathbf{SQ}^{n\times s}, \alpha \in \mathbf{R}$. Then

$$(A+B)^R = A^R + B^R, (\alpha A)^R = \alpha A^R, (AC)^R = A^R C^R.$$

Remark 1. It is noteworthy that $(A^H)^R \not\asymp (A^R)^T$, which is a disadvantage of this real presentation and leads to the inability to deal with generalized inverse and so on.

In [4], Z. Zhang defined other real representation matrix in the form

$$A^r @ \begin{pmatrix} A_1 & A_2 & A_3 & A_4 \\ -A_2 & A_1 & -A_4 & A_3 \\ A_3 & -A_4 & A_1 & -A_2 \\ A_4 & A_3 & A_2 & A_1 \end{pmatrix} \in \mathbf{R}^{4m\times 4n}. \qquad (2.2)$$

A^r has more good properties, but larger scale than A^R. More importantly, A^R is the general matrix and A^r has special structure. In many problems, for example, the following LS question, A^R has more advantages than A^r.

For instance, for $A = A_0 + A_1 i + A_2 j + A_3 k, B = B_0 + B_1 i + B_2 j + B_3 k \in \mathbf{SQ}^{n\times n}$,

$$\begin{aligned} AB &= (A_0B_0 - A_1B_1 + A_2B_2 + A_3B_3) + (A_0B_1 + A_1B_0 - A_2B_3 + A_3B_2)i \\ &\quad + (A_0B_2 + A_2B_0 - A_1B_3 + A_3B_1)j + (A_0B_3 + A_3B_0 + A_1B_2 - A_2B_1)k, \end{aligned}$$

which contains 16 matrix products and 16 matrix additions. But

$$\begin{aligned} A^R B^R &= \begin{pmatrix} A_0 + A_2 & -A_1 + A_3 \\ A_1 + A_3 & A_0 - A_2 \end{pmatrix} \begin{pmatrix} B_0 + B_2 & -B_1 + B_3 \\ B_1 + B_3 & B_0 - B_2 \end{pmatrix} \\ &= \begin{pmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{pmatrix} \begin{pmatrix} B_{11} & B_{12} \\ B_{21} & B_{22} \end{pmatrix} @ \begin{pmatrix} C_{11} & C_{12} \\ C_{21} & C_{22} \end{pmatrix}, \end{aligned}$$

$$AB = \frac{C_{11} + C_{22}}{2} + \frac{C_{21} - C_{12}}{2} i + \frac{C_{11} - C_{22}}{2} j + \frac{C_{21} + C_{12}}{2} k,$$

which contains 8 matrix products and 16 matrix additions.

In [4], for a split quaternion matrix $A = A_0 + A_1 i + A_2 j + A_3 k = (A_0 + A_1 i) + (A_2 + A_3 i) j @ B_0 + B_1 j$ with $B_s \in \mathbf{C}^{m\times n}$, they also defined the following complex representation

$$A^c @ \begin{pmatrix} B_0 & B_1 \\ \bar{B}_1 & \bar{B}_0 \end{pmatrix} \in \mathbf{C}^{2m\times 2n}. \qquad (2.3)$$

A^C has the same size as A^R, but it also has special structure and is a complex matrix. In practical problems, the calculated amount of A^r and A^C is similar and higher than that of A^R.

III. TWO APPLICATIONS

In general, our real representation can deal with many problems without transpose operation and conjugate transpose operation. In this section, as examples, we will study two problems and focus on the least squares problem.

3.1. The split quaternionic least squares problem

For dealing with some problems in the theory and numerical computations of split quaternionic mechanics, one will meet problems of approximate solutions of split quaternion linear equations $AXB \approx E$, that is appropriate when there is error in the matrix E , i.e. split quaternionic least squares (SQLS) problem. The main difficulty in solving this problem is the non-commutative and non-skew-field of the split quaternion and the standard mathematical methods(see [5, 6, 7, 8] and their references) of the complex number field cannot work. In [4], for the first time, the split quaternionic least squares (SQLS) problem was discussed by means of the real representation and the complex representation, which is also main methods for researching the quaternionic least squares (QLS) problem[9, 10, 11, 12, 13, 14]. In this subsection, by means of our real representation, we study the split quaternionic least squares (SQLS) problem, and derive an algebraic techniques for finding solutions of the SQLS problem in split quaternionic mechanics.

Firstly, we discuss the norm of the split quaternion matrix. By the Frobenius norm of complex matrices, we define the following Frobenius norm of the split quaternion matrix $A = A_0 + A_1 i + A_2 j + A_3 k \in \mathbf{SQ}^{m\times n}$.

$$\|A\|_{(F)} \equiv \|A^R\|_F, \qquad (3.1)$$

which has the following properties:

(1) $\|A\|_{(F)} \geq 0$, and equality holds if and only if $A = 0$;

(2) for $\alpha \in \mathbf{R}, \|\alpha A\|_{(F)} = |\alpha| \|A\|_{(F)}$;

(3) $\|A+B\|_{(F)} \leq \|A\|_{(F)} + \|B\|_{(F)}$;

(4) $\|AB\|_{(F)} \leq \|A\|_{(F)} \|B\|_{(F)}$.

(5) $\|A\|_{(F)}^2 = 2(\|A_0\|_F^2 + \|A_1\|_F^2 + \|A_2\|_F^2 + \|A_3\|_F^2)$.

But, different from the Frobenius norm of complex matrices,

(1) for $\alpha \in \mathbf{SQ}, \|\alpha A\|_{(F)} \neq |\alpha| \|A\|_{(F)}$;

(2) $\|A\|_{(F)} \neq \sqrt{trace(A^H A)} = \sqrt{\sum |a_{ij}|^2}$.

Therefore, it is not a natural generality of Frobenius norm for complex matrices, but it is enough to measure the proximity of two split quaternion matrices.

On the basis of the above norm definition, we can give the definition of the split quaternionic least squares (SQLS) problem. Let $A \in \mathbf{SQ}^{m\times n}, B \in \mathbf{SQ}^{p\times q}$, an observation matrix $E \in \mathbf{SQ}^{m\times q}$, and find a matrix $X \in \mathbf{SQ}^{n\times p}$ such that

$$\|AXB - E\|_{(F)} = \min. \qquad (3.2)$$

On the same time, we also construct the following real unconstrained least squares problem

$$\|A^R Y B^R - E^R\|_F = \min, \qquad (3.3)$$

with unknown real matrix Y .

In [4], Z. Zhang studied the SQLS problem (3.2) through the real LS problem (3.3), but here Y has special structure. In fact, Z. Zhang turned the SQLS problem into a real constrained least squares problem.

The following theorem is our main result in this subsection.

Theorem 3.1. Let $A \in \mathbf{SQ}^{m\times n}, B \in \mathbf{SQ}^{p\times q}$, $E \in \mathbf{SQ}^{m\times q}$. Then we have the following results.

(1). The SQLS problem (3.2) has a solution $X \in \mathbf{SQ}^{n\times p}$ if and only if the real LS problem (3.3) has a solution $Y \in \mathbf{R}^{2n\times 2p}$.

(2). If the real LS problem (3.3) has a solution

$$Y = \begin{pmatrix} Y_{11} & Y_{12} \\ Y_{21} & Y_{22} \end{pmatrix} with\, Y_{ij} \in \mathbf{R}^{n\times p}, (i, j = 1, 2),$$

then the solution of the SQLS problem (3.2) can be written as

$$X = \frac{Y_{11}+Y_{22}}{2} + \frac{Y_{21}-Y_{12}}{2} i + \frac{Y_{11}-Y_{22}}{2} j + \frac{Y_{21}+Y_{12}}{2} k. \qquad (3.4)$$

(3).Let $Y = (A^R)^\dagger E (B^R)^\dagger$, which is the minimum norm solution of the real LS problem (3.3) Then the X constructed in (2) is the minimum norm solution of the SQLS problem (3.2).

Proof. (1). If the SQLS problem (3.2) has a solution $X \in \mathbf{SQ}^{n\times p}$, we know that the real matrix X^R is a solution of the real LS problem (3.3). On the other hand, if the real LS problem (3.3) has a solution $Y \in \mathbf{R}^{2n\times 2p}$, then there exit unique matrix set $\{Y_i \in \mathbf{R}^{n\times p}, i = 0,1,2,3\}$ such that

$$Y=\begin{pmatrix} Y_0+Y_2 & -Y_1+Y_3 \\ Y_1+Y_3 & Y_0-Y_2 \end{pmatrix}.$$

So $Y_0+Y_1 i+Y_2 j+Y_3 k\in \mathbf{SQ}^{n\times p}$ is a solution of the SQLS problem (3.2).

(2). The proof of (2) is implicit in the proof of (1).

(3). From

$$\| X \|_{(F)}^2 = \| X^R \|_F^2 = \| Y \|_F^2,$$

we know that (3) is also right.

Based on the above theorem, we obtain the following algorithm.

Algorithm 3.1 Let $A\in \mathbf{SQ}^{m\times n}, B\in \mathbf{SQ}^{p\times q}$, $E\in \mathbf{SQ}^{m\times q}$. Then an algorithm for the solutions of the SQLS problem (3.2) is given as follows.

(1). Construct A^R, B^R and E^R by the definition of realrepresentation (2.1).

(2). Find the solution Y of the real LS problem (3.3).

(3). Construct the solution X of the SQLS problem (3.2):

$$X=\frac{Y(1:n,1:p)+Y(n+1:2n,p+1:2p)}{2}+\frac{Y(n+1:2n,1:p)-Y(1:n,p+1:2p)}{2}i$$
$$+\frac{Y(1:n,1:p)-Y(n+1:2n,p+1:2p)}{2}j+\frac{Y(n+1:2n,1:p)+Y(1:n,p+1:2p)}{2}k.$$

Remark 2. Compared with Algorithm 4.1 of [4], our algorithm 3.1 has only half its size and can greatly reduce the computational space and time complexity. Compared with Algorithm 3.1 of [4], our algorithm 3.1 has the same size, but less computation because their algorithm runs on complex number field.

Remark 3. In Algorithm 3.1, we have many methods for the real LS problem (3.3), such as generalized inverse methods and various iterative methods. Theorem 3.1 contains a generalized inverse method: $Y=(A^R)^{\dagger}E(B^R)^{\dagger}$.

3.2. Inverse matrix

In this subsection, we will give the definition of the inverse matrix and discuss its properties and computation.

Naturally, we give the following definition.

Definition 3.1.Let $A\in \mathbf{SQ}^{n\times n}$. If there is a matrix $B\in \mathbf{SQ}^{n\times n}$ such that

$$AB=BA=I_n,$$

we call A invertible and B is the inverse matrix of A.

Theorem 3.2 Let $A\in \mathbf{SQ}^{n\times n}$. Then A is invertible if and only if A^R is invertible.

Proof. If A is invertible, then there is B such that $AB=BA=I$, which means $A^R B^R=B^R A^R=I^R=I_{2n}$, that is, A^R is invertible.

On the contrary, If A^R is invertible, then there is $B\in \mathbf{R}^{2n\times 2n}$ such that $A^R B=BA^R=I_{2n}$. Partitioned B into

$$B=\begin{pmatrix} B_{11} & B_{12} \\ B_{21} & B_{22} \end{pmatrix}, \text{ with } B_{11}\in \mathbf{R}^{n\times n},$$

then there is a unique set of matrices: B_0, B_1, B_2 and B_3 satisfying

$$B_0+B_2=B_{11}, -B_1+B_3=B_{12}, B_1+B_3=B_{21}, B_0-B_2=B_{22}.$$

Let $\bar{B}=B_0+B_1 i+B_2 j+B_3 k$, we have $\bar{B}^R=B$ and $A^R\bar{B}^R=\bar{B}^R A^R=I_{2n}$, which means $A\bar{B}=\bar{B}A=I_n$, that is, A is invertible.

Algorithm 3.2 Let $A\in \mathbf{SQ}^{n\times n}$. Then an algorithm for the inverse matrix of A is given as follows.

(1). Construct A^R by the definition of realrepresentation (2.1).

(2). Judge the singularity of A^R.

(3). If A^R is nonsingular, find its inverse matrix Y.

(3). Construct the inverse matrix A^{-1} of A:

$$A^{-1}=\frac{Y(1:n,1:p)+Y(n+1:2n,p+1:2p)}{2}+\frac{Y(n+1:2n,1:p)-Y(1:n,p+1:2p)}{2}i$$
$$+\frac{Y(1:n,1:p)-Y(n+1:2n,p+1:2p)}{2}j+\frac{Y(n+1:2n,1:p)+Y(1:n,p+1:2p)}{2}k.$$

Of course, computing $(A^R)^{-1}$ is essentially solving linear matrix equation $A^R Y = I_{2n}$. We have many methods for solving it.

IV. NUMERICAL EXAMPLES

In this section, we give two numerical examples to demonstrate the efficiency of our algorithm. Our examples are performed on an Intel Core i5-6500 3.2GHz/4.00GB computer using Matlab 2016a.

Example 4.1 Given

$$A = \begin{pmatrix} 1+j & 2+k \\ -i & i+j \\ -1+i & k \end{pmatrix}, E = \begin{pmatrix} 1-k & 1+i \\ -j & j+k \\ i+j & 1+k \end{pmatrix}, B = I_2.$$

Find the solution Of SQLS problem (3.2).

This is an example in [4]. Z.Zhang constructed two 12×8 matrices A^r and E^r, found 8×8 matrix Y by solving the real matrix equation $(A^r)^T A^r Y = (A^r)^T E^r$ and then by direct calculation, obtained the unique solutio

$$X_{zhang} = \frac{1}{16}(I_2, -iI_2, jI_2, kI_2)(Y + Q_2^T Y Q_2 + R_2^T Y R_2 + S_2^T Y S_2)\begin{pmatrix} I_2 \\ iI_2 \\ jI_2 \\ kI_2 \end{pmatrix}$$

$$= \begin{pmatrix} 0.5698 - 0.3184i - 0.2179j - 0.8547k & -0.2849 - 0.0279i - 0.1229j - 0.2793k \\ 0.0978 - 0.1788i - 0.0838j - 0.0810k & 0.6006 - 0.0112i + 0.3436j + 0.0503k \end{pmatrix},$$

where Q_2, S_2 and R_2 are defined by [4].

By Algorithm 3.1, we obtain that

$$X_{wang} = \begin{pmatrix} 0.5810 & -0.2849 \\ 0.2709 & 0.6006 \end{pmatrix} + \begin{pmatrix} -0.3128 & -0.0279 \\ -0.0698 & -0.0112 \end{pmatrix} i$$
$$+ \begin{pmatrix} -0.3128 & -0.1229 \\ -0.0698 & 0.3436 \end{pmatrix} j + \begin{pmatrix} -0.5810 & -0.2793 \\ -0.2709 & 0.0503 \end{pmatrix} k,$$

which is not exactly the same as X_{zhang}. Because that

$\| A * X_{zhang} - E \|_{(F)} = 1.9337$ *and* $\| A * X_{wang} - E \|_{(F)} = 1.7877$

we think that, for this example, the result of [4] is doubtful.

Example 4.2 Given $m = n = N, A = A_0 + A_1 i + A_2 j + A_3 k, X = X_0 + X_1 i + X_2 j + X_3 k$ with $A_0 = 10 * rand(m,n), A_1 = 100 * rand(m,n), A_2 = rand(m,n), A_3 = 10 * rand(m,n), X_0 = eye(n), X_1 = zeros(n), X_2 = ones(n),$ $X_3 = zeros(n)$. Let $E = E_0 + E_1 i + E_2 j + E_3 k$ with

$$E_0 = A_0 X_0 - A_1 X_1 + A_2 X_2 + A_3 X_3, E_1 = A_0 X_1 + A_1 X_0 - A_2 X_3 + A_3 X_2,$$
$$E_2 = A_0 X_2 + A_2 X_0 - A_1 X_3 + A_3 X_1, E_3 = A_0 X_3 + A_3 X_0 + A_1 X_2 - A_2 X_1$$

such that $AX = E.$

Based on the generalized inverse, we run Algorithm 4.1 of [4] and our algorithm 3.1, respectively. The former result is written as $XX1$, the latter is written as $XX2$. Table 4.1 gives the results of the calculation.

Table 4.1 Comparison between Algorithm 4.1 of [4] and our algorithm 3.1 with generalize inverse.

N	Algorithm 4.1 of [4]		our algorithm 3.1	
	CPU time	$\frac{\| XX1 - X \|_{(F)}}{\| X \|_{(F)}}$	CPU time	$\frac{\| XX2 - X \|_{(F)}}{\| X \|_{(F)}}$
10	0.0036	4.6e-15	0.0020	8.3e-15
50	0.0089	2.1e-12	0.0044	6.0e-13
100	0.0321	1.7e-12	0.0111	8.1e-13
500	5.3247	7.2e-11	0.4849	5.5e-11
1000	41.974	6.3e-12	5.5684	1.1e-11
2000	-	-	43.100	8.5e-12

In this example, the equation $AX = E$ is consistent and has an unique solution. For the limit of PC, we can not exhibit some examples with large values for N. When $N = 2000$, Algorithm 4.1 of [4] cannot run in our Computer. From Table 4.1, we see that our algorithm 3.1 is better than Algorithm 4.1 of [4].

V. CONCLUSIONS

In this paper, we define a real presentation of the split quaternion matrix, obtain some properties and point out discuss the differences between three representations. As application, we study two problems: the split quaternionic least squares(SQLS) problem and the inverse matrix. Two algebraic methods are obtained and numerical examples show their effectiveness. In general, our real representation can deal with many problems without conjugate transpose operation. Because the generalized inverse is important and involve conjugate transpose operation, how to deal with this issue is worth our further study

Acknowledgements

This work is supported by the National Natural Science Foundation of China(Grant No:11771188)

REFERENCES

[1]. Y. Alagoz, K.H. Oral, S. Yuce, Split quaternion matrices, Miskolc Math. Notes,13(2012)223–232
[2]. D.C. Brody, E.M. Graefe, On complexified mechanics and coquaternions, J. Phys. A: Math. Theor., 44 (2011) 072001.
[3]. N. Moiseyev, Non-Hermitian Quantum Mechanics, Cambridge University Press, 2011.
[4]. Z. Zhang, Z. Jiang, T. Jiang, Algebraic methods for least squares problem in split quaternionic mechanics, Applied Mathematics and Computation, 269 (2015), 618-625.
[5]. Z.-Z. Bai, G.H. Golub and M.K. Ng, Hermitian and skew-Hermitian splitting methods for non-Hermitian positive definite linear systems, SIAM Journal on Matrix Analysis and Applications, 24(2003), 603-626.
[6]. Z.-Z. Bai, M. Benzi and F. Chen, Modified HSS iteration methods for a class of complex symmetric linear systems, Computing, 87(2010), 93-111.
[7]. Z.-Z. Bai, M. Benzi and F. Chen, On preconditioned MHSS iteration methods for complex symmetric linear systems, Numerical Algorithms, 56(2011), 297-317.
[8]. Z.-Z. Bai, Block alternating splitting implicit iteration methods for saddle-point problems from time-harmonic eddy current models, Numerical Linear Algebra with Applications, 19(2012), 914-936.
[9]. M.-H. Wang, An iterative algorithm for least squares problem in quaternionic quantum theory, Comput. Phys. Comm., 179(2008), 203-207.
[10]. M.-H. Wang, On Positive Definite Solutions of Quaternionic Matrix Equations, World Academy of Science, Engineering and Technology, 37(2010), 535-537.
[11]. Q.-W. Wang, X. Zhang, J. W. van der Woude, A new simultaneous decomposition of a matrix quaternity over an arbitrary division ring with applications, Communications in Algebra, 40(2012), 2309-2342.
[12]. 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.
[13]. S.-T Ling, Z.-G Jia, T.-S Jiang, LSQR algorithm with structured preconditioner for the least squares problem in quaternionic quantum theory, Computers and Mathematics with Applications, 73(2017)2208-2220.
[14]. F.-X Zhang, M.-S Wei, Y. Li, J.-L Zhao, Special least squares solutions of the quaternion matrix equation with applications,Applied Mathematics and Computation, 270(2015)425-433.

Minghui Wang "Real Representation and Its Applications in Split Quaternionic Mechanics "IOSR Journal of Mathematics (IOSR-JM) 14.3 (2018): 38-43.