JOURNAL OF GUIDANCE, CONTROL, AND DYNAMICS
Vol. 32, No. 2, March–April 2009

Engineering Notes

ENGINEERING NOTES are short manuscripts describing new developments or important results of a preliminary nature. These Notes should not exceed 2500 words (where a figure or table counts as 200 words). Following informal review by the Editors, they may be published within a few months of the date of receipt. Style requirements are the same as for regular contributions (see inside back cover).

Quaternion Analysis Tools for Engineering and Scientific Applications

James D. Turner*
Texas A&M University, College Station, Texas 77843

DOI: 10.2514/1.35632

I. Introduction

QUATERNION algebra has found a number of applications for engineering and scientific problems, including fluid mechanics [1], quantum mechanics [2,3], robotics [4,5], and spacecraft attitude control [5,6]. Unfortunately, the literature available for supporting engineering applications is diffuse for matrix applications, as well as virtually nonexistent for handling scalar linear and quadratic equations. Very little general-purpose software is available for computing quaternion math models. This Note fills an unmet need for providing a convenient reference for quaternion calculations relevant to engineering and scientific applications. Equally important, the Note provides a compendium of examples and techniques for evaluating scalar linear and quadratic equations, as well as linear matrix equations, matrix inversion, and eigensolutions. Basic operations are defined for all intrinsic operations ($+, -, \times, /, **$) and standard software library function (e.g., sin, cos, ℓn, exp, cosh, a tan, etc.). A comprehensive solution is presented for handling scalar linear equations, in which many special-case solutions are identified. Particular care is given to identifying the equation coefficient conditions required for generating singular solutions. Numerical algorithms are presented for solving scalar quadratic equations. A previously unknown singularity has been identified that has hampered the solution for quadratic equations, and a new algorithm is presented for analytically eliminating the quadratic matrix-valued singularity for solving scalar quadratic equations. Three strategies are presented for handling $N \times N$ matrix operations: 1) purely $N \times N$ quaternion operations, 2) dimension-doubling routines using $2N \times 2N$ complex operations, and 3) dimension-quadrupling $4N \times 4N$ real operations. Many numerical examples are presented in which Quadpack95, a FORTRAN 95/2003 program developed by the author, is used for evaluating all of the quaternion calculations.

Presented as Paper 6160 at the AIAA/AAS Astrodynamics Specialist Conference and Exhibit, Keystone, CO, 21–24 August 2006; received 19 November 2007; revision received 27 February 2008; accepted for publication 28 February 2008.

*Research Professor, Director of Operations, Consortium for Autonomous Space Systems, Department of Aerospace Engineering; turner@aero.tamu.edu.

II. Quaternion Math Models

Sir William Rowan Hamilton invented quaternions [5–10] in 1843, in an effort to generalize complex analysis to 3-dimensional operations, while developing algorithms for rotating his telescope in his observatory. Quaternion algebra combines scalar and vector parts in a single object. The following two forms are used to describe quaternions:

$$\mathsf{a} = a_0 + a_1\hat{\imath} + a_2\hat{\mathbf{j}} + a_3\hat{\mathbf{k}} = (a_0, \mathbf{a})$$

where a_0 denotes the real scalar part of a quaternion, and a_1, a_2, and a_3 denote the real vector components of a quaternion, and

$$\mathbf{a} = a_1\hat{\imath} + a_2\hat{\mathbf{j}} + a_3\hat{\mathbf{k}}$$

denotes the vector part of a quaternion. Mathematically, quaternions form a noncommunitive associative four-dimensional algebra over the real field $\Re$, with an identity element denoted by 1. The orthogonal unit directions of quaternions satisfy the multiplication rules

$$\hat{\imath}^2 = \hat{\mathbf{j}}^2 = \hat{\mathbf{k}}^2 = \hat{\imath}\,\hat{\mathbf{j}}\,\hat{\mathbf{k}} = -1 \tag{1}$$

Basic definitions for $+, -, \times$, and $/$ are defined in Table 1 [7], where $\mathbf{v}_1 \cdot \mathbf{v}_2$ denotes the *vector inner product* and $\mathbf{v}_1 \times \mathbf{v}_2$ denotes the *vector cross product*

Standard mathematical library functions are derived by using addition rules and power-series representations. Addition rules are very useful because they analytically separate the scalar and vector parts of the quaternion. Power-series representations are used for simplifying all vector parts of a quaternion function. As in real and complex analysis, quaternion functions can have one or more valid math models; for example, the sine function has the following two equivalent forms:

$$\sin(q) = \sin(s + \mathbf{v}) = \frac{(\exp(\hat{\boldsymbol{u}}q) - \exp(-\hat{\boldsymbol{u}}q))}{2\hat{\boldsymbol{u}}} \qquad \hat{\boldsymbol{u}} = \frac{\mathbf{v}}{|\mathbf{v}|}$$

where $q = s + \mathbf{v}$ denotes the quaternion and $\hat{\boldsymbol{u}}$ denotes the unit vector for q. The second term is handled by using the addition rule for sine functions and vector trig identities of Appendix A. The third sine function definition deals explicitly with the calculation of a product and division by a vector object. This identity is derived in Appendix B.

Table 1 Basic Math Operations

Math operation	Equation	Quaternion result
Addition	$q_1 + q_2$	$(s_1 + s_2, \mathbf{v}_1 + \mathbf{v}_2)$
Subtraction	$q_1 - q_2$	$(s_1 - s_2, \mathbf{v}_1 - \mathbf{v}_2)$
Multiplication	$q_1 q_2$	$(s_1 s_2 - \mathbf{v}_1 \cdot \mathbf{v}_2, s_1 \mathbf{v}_2 + s_2 \mathbf{v}_1 + \mathbf{v}_1 \times \mathbf{v}_2)$
Conjugation	q^*	$(s, -\mathbf{v})$
Magnitude	$\lvert q \rvert$	$\sqrt{qq^*} = \sqrt{s^2 + \mathbf{v} \cdot \mathbf{v}}$
Division	$1/q$	$q^*/(qq^*)$

A detailed presentation is provided for the exponential function; other series expansions are provided in Table A1. The quaternion exponential calculation is first simplified by introducing the addition rule for exponentials

$$\exp(\mathsf{a}) = \exp(a_0 + \mathbf{a}) = \exp(a_0)\exp(\mathbf{a})$$

which uncouples the scalar and vector parts of the quaternion. The vector-valued exponential function is simplified by introducing the following power series for the scalar exponential as

$$\exp(w) = 1 + w + \frac{w^2}{2!} + \frac{w^3}{3!} + \frac{w^4}{4!} + \cdots \tag{2}$$

The scalar argument for w in Eq. (2) is replaced with the vector argument

$$\mathbf{a} = |\mathbf{a}|\frac{\mathbf{a}}{|\mathbf{a}|} = |\mathbf{a}|\mathbf{u}$$

where $|\mathbf{a}| = \sqrt{(a_1^2 + a_2^2 + a_3^2)}$, leading to

$$\begin{aligned}\exp(|\mathbf{a}|\mathbf{u}) &= 1 + |\mathbf{a}|\mathbf{u} + \frac{|\mathbf{a}|^2}{2!}\mathbf{uu} + \frac{|\mathbf{a}|^3}{3!}\mathbf{uuu} + \cdots\\ &= 1 + |\mathbf{a}|\mathbf{u} - \frac{|\mathbf{a}|^2}{2!} - \frac{|\mathbf{a}|^3}{3!}\mathbf{u} + \cdots;\\ &= \left(\left(1 - \frac{|\mathbf{a}|^2}{2!} + \frac{|\mathbf{a}|^4}{4!} - \cdots\right), \left(|\mathbf{a}| - \frac{|\mathbf{a}|^3}{3!} + \frac{|\mathbf{a}|^5}{5!} - \cdots\right)\mathbf{u}\right)\\ &= (\cos(|\mathbf{a}|), \sin(|\mathbf{a}|)\mathbf{u})\end{aligned}$$

yielding

$$\exp(\mathsf{a}) = \exp(a_0 + \mathbf{a}) = \exp(a_0)(\cos(|\mathbf{a}|), \sin(|\mathbf{a}|)\mathbf{u}) \tag{3}$$

as a straightforward generalization of the exponential for complex functions. Guerlebeck and Sproessig [11] proved the absolute convergence for the quaternion exponential function. The initial Taylor expansion is defined by a tensor equation in $\mathbf{u}$, which is simplified by evaluating the following quaternion product of unit-vector identity:

$$\mathbf{uu} = (0, \mathbf{u})(0, \mathbf{u}) = (0 - \mathbf{u} \cdot \mathbf{u}, 0 + 0 + \mathbf{u} \times \mathbf{u}) = (-1, \mathbf{0}) \tag{4}$$

This unit-vector product identity represents the generalization of the complex-variable identity $\iota^2 = -1$. The application of this identity replaces the quaternion tensorial series expansion with a series consisting of only scalar and vector terms. It is of interest to note that the quaternion exponential of a vector corresponds to the classical Euler parameters commonly used for spacecraft attitude dynamics [5]. See Appendix A for other quaternion identities and mathematical functions.

The main contribution of this Note is the presentation of a systematic approach for deriving and applying quaternion operations for scalar, vector, and matrix operations. Many new results are presented, as well as bringing together many results that have only been available from a very diffuse literature. Scalar linear and quadratic equations are presented in Sec. III. Section IV presents linear equations, matrix inversion, and eigensolutions. Numerical examples are presented in Sec. V, in which the object-oriented operator-overloaded FORTRAN 95/2003 tool known as Quatpack95 provides the numerical solutions. Conclusions are presented in Sec. VI.

III. Linear and Quadratic Quaternion Equations

Quaternion algorithms are presented for solving scalar linear equations and quadratic polynomials. All algorithms explicitly handle the order dependence of terms such as ax and xa. All quaternion components are assumed to be real-valued variables. The necessary conditions for linear and quadratic equations consist of four nonlinear coupled algebraic equations. Linear equations are handled by assembling 4×4 Lyapunov-like equations for inverting the system of four coupled equations [7,8]. This Note introduces a partitioned-solution strategy for the scalar and vector parts of the solution [4]. This approach replaces the original 4×4 Lyapunov-like matrix inversion strategy with a purely linear matrix equation solution. Two steps are required: 1) a closed-form solution is obtained for the scalar part of the solution and 2) the vector part of the solution is obtained by a back-substitution operation. Closed-form expressions are presented for the matrix inverse operator appearing in the partitioned-solution strategy, which provides great insight into the existence of solutions. Many special-solution cases are considered, including coefficient symmetry and antisymmetry, nonsingular general cases, special cases involving symmetry in the coefficients, and singular conditions in which no solution exists.

A. Linear Quaternion Equation: $\mathsf{ax} + \mathsf{xb} + \mathsf{c} = 0$

Two solution methods are presented: 1) the classical 4×4 matrix Lyapunov representation approach and 2) a partitioned-solution method [4] that is based on a purely linear matrix equation approach.

1. *Matrix Lyapunov Equation Method*

Meister [7] and Meister and Schaeben [8] presented the following matrix-valued solution algorithm for handling scalar quaternion equations. Each product term is expanded into four coupled nonlinear equations, and the complete solution is then assembled into a singular matrix Lyapunov-like equation, as follows:

$$\begin{aligned}&\mathsf{ax} + \mathsf{xb} + \mathsf{c} = \mathbf{0} \quad [\mathsf{A}]\mathrm{vec}(\mathsf{x}) + [\mathsf{x}]\mathrm{vec}(\mathsf{b}) = -\mathrm{vec}(\mathsf{c})\\ &[\mathsf{A}]\mathrm{vec}(\mathsf{x}) + [\mathsf{B}']\mathrm{vec}(\mathsf{x}) = -\mathrm{vec}(\mathsf{c}) \quad \underbrace{\mathrm{vec}(\mathsf{x})}_{4\times1} = -\underbrace{\{[\mathsf{A}] + [\mathsf{B}']\}^{-1}}_{4\times4}\underbrace{\mathrm{vec}(\mathsf{c})}_{4\times1}\end{aligned} \tag{5}$$

where $[{}^*]$ denotes a matrix representation for a quaternion, $[{}^{*\prime}]$ denotes matrix representation of a quaternion that accounts for a switch in the product order, and $\mathrm{vec}({}^*)$ denotes a vector representation of the quaternion. The solution for Eq. (5) is reminiscent of the Kronecker product strategies [12] used for solving matrix Lyapunov equations. The solution is robust as long as no singular matrix coefficient conditions are encountered. Solution strategies for identifying and handling singular matrix coefficient conditions are presented in the partitioned-matrix solution section.

2. *Partitioned-Matrix Equation Method*

The necessary condition of Eq. (5) is expanded and the results are collected as follows:

$$\begin{pmatrix}(a_0 + b_0)x_0 - (\mathbf{a} + \mathbf{b}) \cdot \mathbf{x} + c_0\\ (a_0 + b_0)\mathbf{x} + (\mathbf{a} + \mathbf{b})x_0 + (\widetilde{\mathbf{a} - \mathbf{b}}) \cdot \mathbf{x} + \mathbf{c}\end{pmatrix} = \begin{pmatrix}0\\ \mathbf{0}\end{pmatrix} \tag{6}$$

The scalar part of the solution is denoted by the top equation. The vector part of the equation is denoted in the bottom equation. The noncommutivity of product terms is handled by introducing the $\sim$ operator, which denotes the matrix form for the vector cross product given by

$$\tilde{\eta} = \begin{bmatrix}0 & -\eta_3 & \eta_2\\ \eta_3 & 0 & -\eta_1\\ -\eta_2 & \eta_1 & 0\end{bmatrix} \tag{7}$$

The introduction of the $\sim$ operator permits a partitioned-matrix solution strategy to be developed. A full treatment of the solution strategy is only considered for the special case $\mathbf{a} \neq \mathbf{b}$. Five special cases are presented in Table B1, including 1) $\mathbf{a} = \mathbf{b}$, 2) $\mathbf{a} \neq \mathbf{b}$, and 3) $\mathbf{c} = 0$, as well as symmetry and antisymmetry cases for the scalar and vector parts of $\mathbf{a}$, $\mathbf{b}$, and $\mathbf{c}$.

For case $\mathbf{a} \neq \mathbf{b}$, a two-part partitioned-matrix solution strategy is presented. First, the vector part of Eq. (6) is manipulated to provide the following change-of-variable substitution:

$$\mathbf{x} = -[(a_0 + b_0)I_{3\times 3} + \widetilde{\mathbf{a} - \mathbf{b}}]^{-1}(\mathbf{c} + (\mathbf{a} + \mathbf{b})x_0) \tag{8}$$

where $I_{3\times 3}$ denotes a 3×3 identity matrix, the vector cross-product terms are defined by Eq. (7), and $\mathbf{x} = f(x_0)$. Second, introducing Eq. (8) into the scalar part of Eq. (6), one obtains the following closed-form solution for the scalar part of the quaternion linear equation:

$$x_0 = \frac{-\{c_0 + (\mathbf{a} + \mathbf{b}) \cdot [(a_0 + b_0)I_{3\times 3} + \widetilde{\mathbf{a} - \mathbf{b}}]^{-1} \cdot \mathbf{c}\}}{(a_0 + b_0) + (\mathbf{a} + \mathbf{b}) \cdot [(a_0 + b_0)I_{3\times 3} + \widetilde{\mathbf{a} - \mathbf{b}}]^{-1} \cdot (\mathbf{a} + \mathbf{b})} \tag{9}$$

The vector part of the solution is completed by introducing Eq. (9) into Eq. (8). The solution is governed by nonlinear coupling.

Two cases potentially exist in which the solution defined by Eqs. (8) and (9) can fail to exist:

1) The matrix inverse becomes singular.

2) The denominator of Eq. (9) vanishes.

Both cases are investigated. The stability of the matrix inverse solution is investigated by studying the analytic matrix inverse expression given by

$$[(a_0 + b_0)I_{3\times 3} + \widetilde{\mathbf{a} - \mathbf{b}}]^{-1} = \frac{(\mathbf{a} - \mathbf{b})(\mathbf{a} - \mathbf{b})^t/(a_0 + b_0) + (a_0 + b_0)I_{3\times 3} - (\widetilde{\mathbf{a} - \mathbf{b}})}{(\mathbf{a} - \mathbf{b}) \cdot (\mathbf{a} - \mathbf{b}) + (a_0 + b_0)^2} \tag{10}$$

Because the special case of $\mathbf{a} \neq \mathbf{b}$ is being considered, the denominator only vanishes for $\mathbf{b} = -a_0 + \mathbf{a}$. In the numerator, however, Eq. (10) fails to exist when $a_0 + b_0 \Rightarrow 0$ and $\mathbf{b} = -a_0 + \mathbf{a}$.

The second problem is investigated by evaluating the denominator in Eq. (9), given by

$$(a_0 + b_0) + (\mathbf{a} + \mathbf{b}) \cdot [(a_0 + b_0)I_{3\times 3} + \widetilde{\mathbf{a} - \mathbf{b}}]^{-1} \cdot (\mathbf{a} + \mathbf{b}) = 0 \tag{11}$$

when $a_0 + b_0 \neq 0$. Expanding Eq. (11) by introducing the analytic matrix inverse provided by Eq. (10) establishes that the only vanishing solutions for $a_0 + b_0$ are

$$a_0 + b_0 = \pm i\sqrt{(\mathbf{a} - \mathbf{b}) \cdot (\mathbf{a} - \mathbf{b})}$$

which does not represent a physically meaningful quaternion solution (i.e., the quaternion coefficients are not allowed to have complex values). As a result, for the case of $\mathbf{a} \neq \mathbf{b}$, the solution defined by Eqs. (8) and (9) only fails to exist as $a_0 + b_0 \Rightarrow 0$: poor solution behavior is anticipated in some neighborhood of these singular conditions. Five additional special-case scalar linear equation solutions are presented in Table B1.

B. Quadratic Polynomial Equation: $\mathbf{p}^2 + \mathbf{ap} + \mathbf{pb} + \mathbf{c} = 0$

Quadratic equations are more complicated than the previously considered linear equations. Two cases are of interest: 1) $\mathbf{a} = \mathbf{b}$ and 2) $\mathbf{a} \neq \mathbf{b}$. The symmetric case is solved in closed-form solution by completing the square for the polynomial. The general second case is more challenging, because two of the polynomial roots lie very close to a quadratic matrix singularity for the governing equation. Two approaches are presented for handling this case: 1) a simple line search and 2) a homotopy chain method. The quadratic matrix singularity is analytically eliminated by exploiting Laplace's formula for the expansion of the determinate. The transformed problem's necessary condition is solved by 1) introducing a simple linear search algorithm that marches in a direction until a sign change is detected and 2) applying Newton's method for polishing up the line-search root estimates. The second approach develops a continuation-based homotopy chain method for analytically continuing a starting guess solution.

For case 1, $\mathbf{a} = \mathbf{b}$, the quadratic polynomial equation is expressed as $\mathbf{p}^2 + \mathbf{ap} + \mathbf{pa} + \mathbf{c} = 0$, which is solved by completing the square, leading to

$$\mathbf{p}^2 + \mathbf{ap} + \mathbf{pa} + \mathbf{c} = (\mathbf{p} + \mathbf{a})(\mathbf{p} + \mathbf{a}) + \mathbf{c} - \mathbf{a}^2 = 0$$

where the closed-form solution follows as

$$\mathbf{p}^{\pm} = -\mathbf{a} \pm \sqrt{\mathbf{a}^2 - \mathbf{c}} \tag{12}$$

The solution for Eq. (12) is checked by introducing the solution for $\mathbf{p}$ into $\mathbf{p}^2 + \mathbf{ap} + \mathbf{pa} + \mathbf{c} = 0$ and verifying that the symmetric quaternion quadratic equation is satisfied. The quaternion square root is handled as a special case of the general power law given in Appendix A.

For case 2, $\mathbf{a} \neq \mathbf{b}$, the necessary conditions for $\mathbf{p}^2 + \mathbf{ap} + \mathbf{pb} + \mathbf{c} = 0$ are obtained by applying the quaternion product and addition rules and collecting terms, yielding

$$\begin{pmatrix} p_0^2 - \mathbf{p} \cdot \mathbf{p} + (a_0 + b_0)p_0 - (\mathbf{a} + \mathbf{b}) \cdot \mathbf{p} + c_0, \\ 2p_0\mathbf{p} + (a_0 + b_0)\mathbf{p} + (\mathbf{a} + \mathbf{b})p_0 + \widetilde{\mathbf{a} - \mathbf{b}} \cdot \mathbf{p} + \mathbf{c} \end{pmatrix} = \begin{pmatrix} 0 \\ \mathbf{0} \end{pmatrix} \tag{13}$$

Manipulating the last equation provides the nonlinear *partitioned-matrix substitution* for the vector part of the equation as

$$\mathbf{p} = -[\{2p_0 + (a_0 + b_0)\}I_{3\times 3} + \widetilde{\mathbf{a} - \mathbf{b}}]^{-1}(\mathbf{c} + (\mathbf{a} + \mathbf{b})p_0) \tag{14}$$

Introducing Eq. (14) into the scalar equation of Eq. (13),

$$p_0^2 + (a_0 + b_0)p_0 - (\mathbf{a} + \mathbf{b}) \cdot \mathbf{p} - \mathbf{p} \cdot \mathbf{p} + c_0 = 0 \tag{15}$$

yields a single nonlinear polynomial equation for the scalar part of the necessary condition.

1. *Numerical Solution Strategy for Polynomial Roots Near a Quadratic Matrix Singularity*

Plots of Eq. (15) indicate that two roots of the equation are located in the vicinity of $p_0 = -(a_0 + b_0)/2$, which corresponds to a quadratic singularity for the p_0 polynomial of Eq. (15). The singular behavior of Eq. (15) is eliminated by exploiting Laplace's formula for the expansion of the determinate of an $n \times n$ matrix, given by

$$A \times \mathrm{adj}(A) = \det(A)I_{n\times n} \Rightarrow A^{-1} = \mathrm{adj}(A)/\det(A)$$

where adj(*) denotes the classical adjoint, and det(*) denotes the matrix determinant. As shown in what follows, this matrix identity allows the singular behavior of the determinate of the matrix inverse to be explicitly factored out of the equation. The matrix of Eq. (14) is defined as

$$M = \{2p_0 + (a_0 + b_0)\}I_{3\times 3} + \widetilde{\mathbf{a} - \mathbf{b}} \tag{16}$$

where the determinant and adjoint are given by

$$\det(M) = (2p_0 + a_0 + b_0)((\mathbf{a} - \mathbf{b}) \cdot (\mathbf{a} - \mathbf{b}) + (2p_0 + a_0 + b_0)^2)$$

$$\mathrm{adjoint}(M) = (\mathbf{a} - \mathbf{b})(\mathbf{a} - \mathbf{b})^t + (2p_0 + a_0 + b_0)^2 I_{3\times 3} - (2p_0 + a_0 + b_0)(\widetilde{\mathbf{a} - \mathbf{b}})$$

where the determinant is easily seen as the cause of the singularity condition. Equation (15) is transformed to a singularity-*free* form by introducing Eq. (14) and multiplying the resulting equation by

$\det(M)^2$, yielding

$$\det(M)^2(p_0^2 + (a_0 + b_0)p_0 + c_0) - \det(M)(\mathbf{a} + \mathbf{b}) \cdot \mathbf{y} - \mathbf{y} \cdot \mathbf{y} = 0$$
$$\mathbf{y} = -\text{adjoint}(M)(\mathbf{c} + (\mathbf{a} + \mathbf{b})p_0) \tag{17}$$

Plots of Eq. (17) indicate that the roots are symmetrically located relative to p_0, where symmetry has been observed to be 5–8 digits relative to the centerline of the quadratic well. Starting with the well centerline location defined by $p_0 = -(a_0 + b_0)/2$, a simple line-search algorithm refines the root estimate. The analyst selects a search direction, defines a step size in Δp_0, and evaluates Eq. (17) until a sign change is detected on the kth step of the process. An interpolated root estimate is obtained by fitting a linear line model to the last two data points for p_0:

$$p_{0,k} \approx -\frac{y_{k-1} - m p_{0,k-1}}{m}$$

$$m = \frac{y_k - y_{k-1}}{p_{0,k} - p_{0,k-1}}$$

Application of a finite-difference-based Newton iteration yields *10-place* accuracy after 5–6 iterations. The second root is found by exploiting the problem symmetry. The second root is estimated to be $p_{0,k_2} \approx -2x_0 - p_{0,k_2}$ and refined with Newton's method. The polished root values are introduced into Eq. (14) to complete the vector part of the solution for the quaternion roots of $\mathbf{p}^2 + \mathbf{ap} + \mathbf{pb} + \mathbf{c} = 0$.

2. *Homotopy Integration Method*

The section presents an alternative root-solving approach for quadratic quaternion equations. The basic idea is to use analytic continuation to transform a known solution for a neighboring quadratic equation into the solution for the desired quadratic equation. The *completing-the-square* solution of Eq. (12) provides a starting solution for introducing a homotopy parameter that redefines the quadratic equation as

$$\mathbf{P}^2 + \mathbf{aP} + \mathbf{P}(\mathbf{a} + s(\mathbf{b} - \mathbf{a})) + \mathbf{c} = 0 \qquad \mathbf{P} = \mathbf{P}(s)$$

where s denotes the homotopy parameter that is varied from $s = 0$ to 1. Evaluating the polynomial at $s = 0$ leads to the solution defined by Eq. (12). Setting $s = 1$ yields the solution for the desired quadratic solutions. Differentiating the polynomial with regard to s yields the quaternion differential equation:

$$\mathbf{f}\,\mathbf{P}_{,\mathbf{s}} + \mathbf{P}_{,\mathbf{s}}\mathbf{d} = \mathbf{e} \quad \text{subject to } \mathbf{P}^{\pm}|_{s=0} = -\mathbf{a} \pm \sqrt{\mathbf{a}^2 - \mathbf{c}}$$

where $\mathbf{f} = \mathbf{P} + \mathbf{a}$, $\mathbf{d} = \mathbf{P} + \mathbf{a} + s(\mathbf{b} - \mathbf{a})$, and $\mathbf{e} = \mathbf{P}(\mathbf{b} - \mathbf{a})$. The quaternion differential equation is solved as a scalar linear quaternion equation of the type previously considered for the derivative value as a function of s. The quadratic roots are obtained by numerically integrating the equation

$$\mathbf{P}|_{s=1} = \mathbf{P}|_{s=0} + \int_0^1 \mathbf{P}_{,\mathbf{s}} \mathrm{d}s$$

(fourth-order Runge–Kutta algorithm using ~50 steps yields ~10 digits). This algorithm provides a powerful way to solve for the roots, assuming that the solution for the linear differential equation remains well-behaved along the homotopy path.

IV. Quaternion Matrix Equations

Three classes of quaternion matrix equations are presented in this section, including 1) a linear matrix equation, 2) matrix inversion, and 3) eigenvalue problems. Three approaches are presented for solving quaternion matrix equations in $Q^{N\times N}$: 1) pure quaternion-based algorithms in $Q^{N\times N}$, 2) dimension-doubling to create a complex matrix in $\mathbb{C}^{2N\times 2N}$, and 3) dimension-quadrupling to create a real matrix in $\Re^{4N\times 4N}$. The order dependence of linear algebra operations is minimized by solving equations in higher-dimensional complex and real-dimensioned spaces. It is well known that linear matrix equations and matrix inversion algorithms can both be solved by using purely quaternion-based algorithms [13–19], though the literature is often very diffuse and focused on theoretical rather than computational issues. Nonlinear eigenvalue problems [15,16] are handled by generating a complex matrix of twice the dimension. A further complication with quaternion eigenvalue problems is that both right and left eigenvalue problems exist, because of the order dependence of quaternion products [20]. The results in this Note are restricted to right eigenvalue problems.

A. Linear Matrix Equations

Let a quaternion linear matrix equation be defined by $\mathbf{Ax} = \mathbf{b}$, where $\mathbf{A} \in \mathbb{Q}^{N\times N}$ and $\mathbf{x}, \mathbf{b} \in \mathbb{Q}^{N\times 1}$. As long as the order dependence of operations is respected, quaternion matrix algorithms are easily obtained from real or complex-variable counterparts. For example, Gaussian elimination, lower/upper factorization, quadratic-regulator factorization, and others all work using quaternion algebra. The order dependence of operations is addressed by considering the following row-reduction operation:

$$a_{j1}x_1 + a_{j1}x_2 + \cdots + a_{jm}x_m = b_j$$
$$a_{k1}x_1 + a_{k1}x_2 + \cdots + a_{km}x_m = b_k$$

During a typical triangularization step, one seeks to add

$$-\frac{a_{k1}}{a_{j1}} \times (\text{first equation})$$

to the second equation, leading to the transformed set of equations:

$$\begin{aligned} &a_{j1}x_1 + a_{j2}x_2 + \cdots + a_{jm}x_m = b_j \\ &\times\left(a_{k2} - \left(\frac{a_{k1}}{a_{j1}}\right) \times a_{j2}\right)x_2 + \cdots + \left(a_{km} - \left(\frac{a_{k1}}{a_{j1}}\right) \times a_{jm}\right)x_m \\ &= \left(b_k - \left(\frac{a_{k1}}{a_{j1}}\right) \times b_j\right) \end{aligned} \tag{18}$$

where the leading term of the second equation vanishes and the remaining terms are easily recognized as *Gaussian elimination updates*. The quaternion division operation is assumed to be defined as being equivalent to $\mathbf{a}/\mathbf{b} = \mathbf{a} \times (1/\mathbf{b})$. As a specific example, given the three quaternions $\mathbf{a} = 1 + 2\hat{\imath} + 3\hat{\mathbf{j}} + 4\hat{\mathbf{k}}$, $\mathbf{b} = -1 + 3\hat{\imath} + \hat{\mathbf{j}} - 4\hat{\mathbf{k}}$, and $\mathbf{c} = 3 - \hat{\imath} - 2\hat{\mathbf{j}} + \hat{\mathbf{k}}$ and performing the operations indicated in the Gaussian elimination update equations as

$$\left(\mathbf{a}/\mathbf{b}\right) \times \mathbf{c} = -\frac{5}{2} + \frac{1}{2}\hat{\imath} - \frac{5}{2}\hat{\mathbf{j}} - \frac{3}{2}\hat{\mathbf{k}}$$
$$(\mathbf{a} \times \mathbf{c})/\mathbf{b} = -\frac{13}{30} - \frac{37}{30}\hat{\imath} - \frac{109}{30}\hat{\mathbf{j}} - \frac{3}{10}\hat{\mathbf{k}}$$

the results are not the same. The top equation is correct. The bottom equation is wrong, but this form of the update equation frequently appears in texts for linear algebra. Quaternion Gaussian elimination requires that the first operation is division.

Quaternion matrix inversion routines are easily developed by replacing the right-hand side with the appropriate columns of the identity matrix and generating n-solution vectors to construct the inverse matrix.

Quaternion algebra is avoided in matrix operations by defining a complex matrix of twice the dimension of the quaternion matrix. The transformation is carried out by manipulating the following equations:

$$\underbrace{\mathbf{A}}_{n\times n}\underbrace{\mathbf{x}}_{n\times 1}=\underbrace{\mathbf{b}}_{n\times 1}$$

$$\begin{aligned}&[A_0+A_1\hat{\imath}+A_2\hat{\mathbf{j}}+A_3\hat{\mathbf{k}}](x_0+x_1\hat{\imath}+x_2\hat{\mathbf{j}}+x_3\hat{\mathbf{k}})\\&=b_0+b_1\hat{\imath}+b_2\hat{\mathbf{j}}+b_3\hat{\mathbf{k}}[(A_0+A_1\hat{\imath})+(A_2+A_3\hat{\imath})\hat{\mathbf{j}}]((x_0+x_1\hat{\imath})\\&+(x_2+x_3\hat{\imath})\hat{\mathbf{j}})=(b_0+b_1\hat{\imath})+(b_2+b_3\hat{\imath})\hat{\mathbf{j}}[M_1+M_2\hat{\mathbf{j}}]\\&\times(y_1+y_2\hat{\mathbf{j}})=d_1+d_2\hat{\mathbf{j}}\end{aligned}\tag{19}$$

where $\mathbf{A}\in\mathbb{Q}^{N\times N}$; $\mathbf{x},\mathbf{b}\in\mathbb{Q}^{N\times 1}$; M_1 and $M_2\in\mathbb{C}^{N\times N}$; and y_1, y_2, d_1, and $d_2\in\mathbb{C}^{N\times 1}$. The dimension-doubling algorithm is obtained by manipulating the final equation, leading to

$$\left.\begin{aligned}&[M_1+M_2\hat{\mathbf{j}}](y_1+y_2\hat{\mathbf{j}})=d_1+d_2\hat{\mathbf{j}}\\&M_1y_1+M_1y_2\hat{\mathbf{j}}+M_2\hat{\mathbf{j}}y_1+M_2\hat{\mathbf{j}}y_2\hat{\mathbf{j}}=d_1+d_2\hat{\mathbf{j}}\\&M_1y_1+M_1y_2\hat{\mathbf{j}}+M_2y_1^*\hat{\mathbf{j}}-M_2y_2^*=d_1+d_2\hat{\mathbf{j}}\\&M_1y_1+M_2(-y_2^*)-M_2^*y_1\hat{\mathbf{j}}+M_1^*(-y_2^*)\hat{\mathbf{j}}=d_1-d_2^*\hat{\mathbf{j}}\end{aligned}\right\}\tag{20}$$

$$\Rightarrow\underbrace{\begin{bmatrix}M_1 & M_2\\-M_2^* & M_1^*\end{bmatrix}}_{2n\times 2n}\underbrace{\begin{pmatrix}y_1\\-y_2^*\end{pmatrix}}_{2n\times 1}=\underbrace{\begin{pmatrix}d_1\\-d_2^*\end{pmatrix}}_{2n\times 1}$$

where the multiplication rules have been applied several times. The intermediate results are most easily obtained by expressing $\hat{\imath}$ and $\hat{\mathbf{j}}$ as quaternions in the negation and conjugation steps in Eq. (20). For example, consider the simplification of the triple product

$$\begin{aligned}\hat{\mathbf{j}}y_2\hat{\mathbf{j}}&=(0,\hat{\mathbf{j}})(b_2,b_3\hat{\imath})(0,\hat{\mathbf{j}})=(0,\hat{\mathbf{j}})(0,b_2\hat{\mathbf{j}}+b_3\hat{\mathbf{k}})\\&=(-b_2,b_3\hat{\imath})=-y_2^*\end{aligned}$$

Equation (20) is solved as a complex-valued matrix equation for y_1 and y_2. The quaternion solution is recovered as

$$\begin{aligned}\mathbf{x}&=y_1+y_2\hat{\mathbf{j}}=x_0+x_1\hat{\imath}+x_2\hat{\mathbf{j}}+x_3\hat{\mathbf{k}}\\\mathbf{x}&=\mathrm{Re}(y_1)+\mathrm{Im}(y_1)\hat{\imath}+\mathrm{Re}(y_2)\hat{\mathbf{j}}+\mathrm{Im}(y_2)\hat{\mathbf{k}}\end{aligned}\tag{21}$$

B. Quaternion Eigenvalue and Eigenvector Equation

Quaternion eigenvalues and eigenvectors are obtained for matrix $\mathbf{A}\in\mathbb{Q}^{N\times N}$. As shown by Brenner [14] and Zhang [13], any $N\times N$ quaternion matrix has exactly N (right) eigenvalues that are complex numbers. The right eigenvalue problem is defined by $\mathbf{Ax}=\mathbf{x}\lambda$ and the left eigenvalue problems is defined by $\mathbf{Ax}=\lambda\mathbf{x}$. Expressing $\mathbf{A}=A_1+A_2\hat{\mathbf{j}}$, $\mathbf{x}=x_1+x_2\hat{\mathbf{j}}$, and $\lambda=\lambda_1+\lambda_2\hat{\imath}$, where A_1 and $A_2\in\mathbb{C}^{N\times N}$, x_1 and $x_2\in\mathbb{C}^{N\times 1}$, and λ_1 and $\lambda_2\in\mathbb{C}$, then the eigenvalue problem can be shown to be equivalent to

$$\begin{bmatrix}A_1 & A_2\\-A_2^* & A_1^*\end{bmatrix}\begin{pmatrix}x_1\\-x_2^*\end{pmatrix}=\lambda\begin{pmatrix}x_1\\-x_2^*\end{pmatrix}\tag{22}$$

or

$$\begin{bmatrix}A_1 & A_2\\-A_2^* & A_1^*\end{bmatrix}\begin{pmatrix}x_2\\x_1^*\end{pmatrix}=\lambda^*\begin{pmatrix}x_2\\x_1^*\end{pmatrix}\tag{23}$$

For Eq. (22) the eigenvectors are recovered as follows:

$$\begin{aligned}\mathbf{x}&=x_1+x_2\hat{\mathbf{j}}=a+b\hat{\imath}+c\hat{\mathbf{j}}+d\hat{\mathbf{k}}\\\mathbf{x}&=\mathrm{Re}(x_1)+\mathrm{Im}(x_1)\hat{\imath}+\mathrm{Re}(x_2)\hat{\mathbf{j}}+\mathrm{Im}(x_2)\hat{\mathbf{k}}\end{aligned}\tag{24}$$

Similar results follow when Eq. (23) is used. Left eigenvalue problems, defined by $\mathbf{Ax}=\lambda\mathbf{x}$, are extremely difficult problems [20] and are not considered in this Note. One needs to be aware of the nonuniqueness of quaternion eigenvalues. The eigenvalues are unique only up to a conjugation by another quaternion [i.e., if $\mathbf{A}$ is a quaternion matrix with eigenvalue lambda and eigenvector $\mathbf{x}$ (i.e., if $\mathbf{Ax}=\mathbf{x}\lambda$)], then for all nonzero quaternions $\mathbf{w}$,

$$\mathbf{A}\,(\mathbf{xw})=(\mathbf{xw})\{(1/\mathbf{w})\lambda\mathbf{w}\}$$

As a result, $(1/\mathbf{w})\lambda\mathbf{w}$ is also an eigenvalue with eigenvector $\mathbf{xw}$. It is not unusual to choose a preferred unique representative eigenvalue [e.g., where the value of $(1/\mathbf{w})\lambda\mathbf{w}$ is located in the upper-half complex plane]. Equations (22) and (23) are in the standard form for a general complex-valued eigensolution. The application section presents numerical results in which MATLAB is used.

V. Applications

Numerical results are provided for the scalar linear equations of Sec. III. The following quaternion parameters are assumed for all applications:

$$\begin{gathered}\mathbf{a}=1-2\hat{\imath}+\hat{\mathbf{j}}+2\hat{\mathbf{k}}\qquad\mathbf{b}=-2+3\hat{\imath}+2\hat{\mathbf{j}}+\hat{\mathbf{k}}\\\mathbf{c}=-1-2\hat{\imath}+3\hat{\mathbf{j}}-4\hat{\mathbf{k}}\end{gathered}\tag{25}$$

The results presented in Table B2 display four digits of numerical precision; however, all calculations have been performed to double-precision to assess the solution accuracy.

A. Linear and Quadratic Equation Examples

Table B1 presents the equation type, the assumed coefficient properties, the resulting solution, and the accuracy achieved for each component of the quaternion. The first linear case in Table B2 represents case 1 in Table B1, in which $\mathbf{a}=\mathbf{b}$, $\mathbf{a}_0\neq 0$, and $\mathbf{c}\neq 0$. This solution is defined by closed-form expression, which provides extremely high accuracy. The second linear case in Table B2 corresponds to the general solution defined by Eqs. (8) and (9), in which $\mathbf{a}\neq\mathbf{b}$, $\mathbf{a}_0+\mathbf{b}_0\neq\mathbf{0}$, and $\mathbf{c}\neq 0$. Four additional special-case solutions are identified in Table B1.

In Table B1, when $\mathbf{a}=\mathbf{b}$, the solution is obtained from the completing-the-square algorithm of Eq. (12). The quaternion square-root algorithm following Eq. (12) is used and high accuracy is obtained. The last quadratic case in Table B1, in which $\mathbf{a}\neq\mathbf{b}$, makes use of Eq. (17) to avoid the quadratic matrix singularity. After transforming the original scalar polynomial equation, using the singular value for the original matrix, and conducting the linear search algorithm for detecting a sign change in the polynomial, the final root value is obtained by using Newton's method. In all cases, the solution accuracy achieved is very high.

B. Matrix Equation Examples

Two matrix examples are considered: 1) a 2×2 matrix inverse and 2) a 3×3 linear equation solution. The first example consists of a 2×2 matrix inverse, which is obtained by using the Gaussian elimination for quaternions. The assumed quaternion matrix is given by

$$\underbrace{\mathbf{A}}_{2\times 2}=\begin{bmatrix}1 & \hat{\mathbf{j}}\\\hat{\mathbf{k}} & 2\end{bmatrix}$$

The matrix is augmented by a 2×2 identity matrix and transformed as

$$\begin{aligned}B=\begin{bmatrix}1 & \hat{\mathbf{j}} & 1 & 0\\\hat{\mathbf{k}} & 2 & 0 & 1\end{bmatrix}&\Rightarrow\begin{bmatrix}1 & \hat{\mathbf{j}} & 1 & 0\\0 & 2+\hat{\imath} & -\hat{\mathbf{k}} & 1\end{bmatrix}\\&\Rightarrow\begin{bmatrix}1 & 0 & 1+\hat{\mathbf{j}}\frac{1}{2+\hat{\imath}}\hat{\mathbf{k}} & -\hat{\mathbf{j}}\frac{1}{2+\hat{\imath}}\\0 & 1 & -\frac{1}{2+\hat{\imath}}\hat{\mathbf{k}} & \frac{1}{2+\hat{\imath}}\end{bmatrix}\end{aligned}$$

using standard Gaussian elimination operations, in which the order of operations is preserved. The quaternion matrix inverse follows as

$$\begin{aligned}\underbrace{\mathbf{A}^{-1}}_{2\times 2}&=\begin{bmatrix}1+\hat{\mathbf{j}}\frac{1}{2+\hat{\imath}}\hat{\mathbf{k}} & -\hat{\mathbf{j}}\frac{1}{2+\hat{\imath}}\\-\frac{1}{2+\hat{\imath}}\hat{\mathbf{k}} & \frac{1}{2+\hat{\imath}}\end{bmatrix}\\&=\begin{bmatrix}0.8+0.4\hat{\imath} & -0.4\hat{\mathbf{j}}-0.2\hat{\mathbf{k}}\\-0.2\hat{\mathbf{j}}-0.4\hat{\mathbf{k}} & 0.2-0.2\hat{\imath}\end{bmatrix}\end{aligned}$$

which is easily verified by computing the product of the matrix and its inverse.

A 3×3 linear matrix equation is numerically solved using a purely quaternion-based algorithm. Only four digits of the solution are presented. However, the calculation has been performed to double-precision accuracy and a machine-accurate result is obtained. The input quaternion matrix and right-hand side are defined by

$$\underbrace{\mathbf{A}}_{3\times 3} = \begin{bmatrix} (1\ \ 2\ \ 3\ \ 4) & (2\ \ -1\ \ 5\ \ 10) & (-3\ \ 9\ \ 4\ \ -2) \\ (6\ \ 2\ \ -3\ \ 1) & (4\ \ 4\ \ 1\ \ 1) & (-1\ \ -10\ \ 2\ \ 1) \\ (4\ \ -3\ \ -1\ \ 5) & (8\ \ -6\ \ 9\ \ 2) & (-10\ \ 2\ \ 4\ \ 11) \end{bmatrix}$$

$$\underbrace{\mathbf{b}}_{3\times 1} = \begin{bmatrix} (1\ \ 1\ \ 1\ \ 1) \\ (2\ \ 4\ \ -2\ \ -4) \\ (-5\ \ -3\ \ -2\ \ 1) \end{bmatrix}$$

The solution for the linear system is given by

$$\underbrace{\mathbf{x}}_{3\times 1} = \begin{bmatrix} (0.75758, 0.32527, -1.08560, -0.28725) \\ (0.07264, -0.03779, 0.53136, -0.18380) \\ (0.11327, 0.17580, 0.33267, 0.22560) \end{bmatrix}$$

where the solution accuracy has been checked by computing the error equation (error $= Ax - b$), in which the largest error is $\sim 10^{-15}$.

C. Quaternion Eigensolution Problem

A single quaternion eigenvalue/eigenvector example has been run. Only a right eigenvalue problem is evaluated. The quaternion matrix is

$$\mathbf{A} = \begin{bmatrix} 0 & 1-\hat{\mathbf{k}} \\ 1-\hat{\mathbf{k}} & 0 \end{bmatrix}$$

(This example has been suggested to the author during a private communication with Ralph Byers, Mathematics Department, Kanus State University.) The companion complex-valued matrix of Eq. (22) is given by

$$\mathbf{C} = \begin{bmatrix} 0 & 1 & 0 & -\hat{\imath} \\ 1 & 0 & -\hat{\imath} & 0 \\ 0 & -\hat{\imath} & 0 & 1 \\ -\hat{\imath} & 0 & 1 & 0 \end{bmatrix}$$

leading to the MATLAB-generated eigensolution given by

$$V = \begin{bmatrix} 1 & -1 & 1 & 1 \\ -1 & 1 & 1 & 1 \\ 1 & 1 & -1 & 1 \\ -1 & -1 & -1 & 1 \end{bmatrix}$$

$$D = \begin{bmatrix} -1+\hat{\imath} & -1-\hat{\imath} & 1+\hat{\imath} & 1-\hat{\imath} \end{bmatrix}$$

where V denotes the eigenvectors and D denotes the eigenvalues. Transforming the 4×4 results for the 2×2 case leads to

$$\mathbf{v}_1 = \begin{pmatrix} 1+\hat{\mathbf{j}} \\ 1+\hat{\mathbf{j}} \end{pmatrix} \quad d_1 = 1+\hat{\imath} \quad \mathbf{v}_2 = \begin{pmatrix} 1-\hat{\mathbf{j}} \\ -1+\hat{\mathbf{j}} \end{pmatrix} \quad d_2 = -1+\hat{\imath}$$

which can be verified by direct computation of $\mathbf{A}\mathbf{v}_i = \mathbf{v}_i d_i$ $(i = 1, 2)$. Complex conjugate eigenvalues of $\mathbf{C}$ correspond to the same quaternion eigenvalue of $\mathbf{A}$, as shown by

$$1-\hat{\imath} = (-1/\hat{\mathbf{j}})(1+\hat{\imath})(-\hat{\mathbf{j}}) \qquad -1-\hat{\imath} = (1/\hat{\mathbf{j}})(-1+\hat{\imath})(\hat{\mathbf{j}})$$

VI. Conclusions

The theoretical foundations for quaternion calculations for engineering and scientific applications are presented. Many of the results are not commonly available in the engineering literature. Complete derivations are presented for all of the intrinsic and math library functions. New results are presented for scalar linear equations, polynomial equations, and matrix linear and inversion algorithms. Strategies for handling matrix equations are presented that use purely quaternion operations as well as dimension-doubling operations using complex operations. Numerical examples are presented for demonstrating each capability.[†] This Note provides a convenient reference for advanced quaternion analysis methods as well as a compendium of examples and techniques for analysis methods relevant to real-world applications in science and engineering.

Appendix A: Quaternion Library Functions

This Appendix summarizes all quaternion forms for mathematical library functions. Power-series expansions are used for deriving all results except for inverse functions. Frequent use is made of addition rules that allow the quaternion calculation to be separated into scalar and vector parts. The first three equations deal with the calculation for functions of vectors, because these identities are required for simplifying many of the following identities.

I. Trig and Exponential Functions of Vectors

$$\cos(\mathbf{a}) = 1 - \frac{\mathbf{aa}}{2!} + \frac{\mathbf{aaaa}}{4!} - \cdots = 1 - \frac{|\mathbf{a}|^2}{2!}\frac{\mathbf{a}}{|\mathbf{a}|}\frac{\mathbf{a}}{|\mathbf{a}|} + \frac{|\mathbf{a}|^4}{4!}\frac{\mathbf{a}}{|\mathbf{a}|}\frac{\mathbf{a}}{|\mathbf{a}|}\frac{\mathbf{a}}{|\mathbf{a}|}\frac{\mathbf{a}}{|\mathbf{a}|} - \cdots = 1 + \frac{|\mathbf{a}|^2}{2!} + \frac{|\mathbf{a}|^4}{4!} + \cdots = \cosh(|\mathbf{a}|)$$

$$\sin(\mathbf{a}) = \mathbf{a} - \frac{\mathbf{aaa}}{3!} + \frac{\mathbf{aaaaa}}{5!} - \cdots = |\mathbf{a}|\frac{\mathbf{a}}{|\mathbf{a}|} - \frac{|\mathbf{a}|^3}{3!}\frac{\mathbf{a}}{|\mathbf{a}|}\frac{\mathbf{a}}{|\mathbf{a}|}\frac{\mathbf{a}}{|\mathbf{a}|} + \frac{|\mathbf{a}|^5}{5!}\frac{\mathbf{a}}{|\mathbf{a}|}\frac{\mathbf{a}}{|\mathbf{a}|}\frac{\mathbf{a}}{|\mathbf{a}|}\frac{\mathbf{a}}{|\mathbf{a}|}\frac{\mathbf{a}}{|\mathbf{a}|} - \cdots = \left(|\mathbf{a}| + \frac{|\mathbf{a}|^2}{2!} + \frac{|\mathbf{a}|^4}{4!} + \cdots\right)\frac{\mathbf{a}}{|\mathbf{a}|} = \sinh(|\mathbf{a}|)\frac{\mathbf{a}}{|\mathbf{a}|}$$

$$\exp(\mathbf{a}) = \exp\left(|\mathbf{a}|\frac{\mathbf{a}}{|\mathbf{a}|}\right) = 1 + |\mathbf{a}|\frac{\mathbf{a}}{|\mathbf{a}|} + \frac{|\mathbf{a}|^2}{2!}\frac{\mathbf{a}}{|\mathbf{a}|}\frac{\mathbf{a}}{|\mathbf{a}|} + \frac{|\mathbf{a}|^3}{3!}\frac{\mathbf{a}}{|\mathbf{a}|}\frac{\mathbf{a}}{|\mathbf{a}|}\frac{\mathbf{a}}{|\mathbf{a}|} + \frac{|\mathbf{a}|^4}{4!}\frac{\mathbf{a}}{|\mathbf{a}|}\frac{\mathbf{a}}{|\mathbf{a}|}\frac{\mathbf{a}}{|\mathbf{a}|}\frac{\mathbf{a}}{|\mathbf{a}|} + \frac{|\mathbf{a}|^5}{5!}\frac{\mathbf{a}}{|\mathbf{a}|}\frac{\mathbf{a}}{|\mathbf{a}|}\frac{\mathbf{a}}{|\mathbf{a}|}\frac{\mathbf{a}}{|\mathbf{a}|}\frac{\mathbf{a}}{|\mathbf{a}|} + \cdots = 1 + |\mathbf{a}|\frac{\mathbf{a}}{|\mathbf{a}|} - \frac{|\mathbf{a}|^2}{2!} - \frac{|\mathbf{a}|^3}{3!}\frac{\mathbf{a}}{|\mathbf{a}|} + \frac{|\mathbf{a}|^4}{4!} - \frac{|\mathbf{a}|^5}{5!}\frac{\mathbf{a}}{|\mathbf{a}|} + \cdots = \left(1 - \frac{|\mathbf{a}|^2}{2!} + \frac{|\mathbf{a}|^4}{4!} - \cdots\right) + \left(|\mathbf{a}| - \frac{|\mathbf{a}|^3}{3!} + \frac{|\mathbf{a}|^5}{5!} - \cdots\right)\frac{\mathbf{a}}{|\mathbf{a}|} = \left(\cos(|\mathbf{a}|, \sin(|\mathbf{a}|)\frac{\mathbf{a}}{|\mathbf{a}|}\right)$$

II. Basic Simplification Identity

$$\frac{\mathbf{a}}{|\mathbf{a}|}\mathbf{a} = \frac{\mathbf{a}}{|\mathbf{a}|}(a_0 + \mathbf{a}) = -|\mathbf{a}| + a_0\frac{\mathbf{a}}{|\mathbf{a}|}$$

[†]All calculations have been performed with the Fortran 95/2003 Quatpack95 toolbox that incorporates all of the theoretical capabilities presented here. Quatpack95 is available by contacting the Author.

Table A1 Quaternion Math Library

Function Name	Math model
Cosine	$\cos(\mathbf{a}) = \cos(a_0 + \mathbf{a}) = \cos(a_0)\cos(\mathbf{a}) - \sin(a_0)\sin(\mathbf{a}) = (\cos(a_0)\cosh(\lvert\mathbf{a}\rvert), -\sin(a_0)\sinh(\lvert\mathbf{a}\rvert)\frac{\mathbf{a}}{\lvert\mathbf{a}\rvert})$
Sine	$\sin(\mathbf{a}) = \sin(a_0 + \mathbf{a}) = \sin(a_0)\cos(\mathbf{a}) + \cos(a_0)\sin(\mathbf{a}) = (\sin(a_0)\cosh(\lvert\mathbf{a}\rvert), \cos(a_0)\sinh(\lvert\mathbf{a}\rvert)\frac{\mathbf{a}}{\lvert\mathbf{a}\rvert})$
Tangent	$\tan(\mathbf{a}) = \sin(\mathbf{a})/\cos(\mathbf{a})$
Arc cosine	$a\cos(\mathbf{a}) = -\frac{\mathbf{a}}{\lvert\mathbf{a}\rvert}\, a\cosh(\mathbf{a})$
Arc sine	$a\sin(\mathbf{a}) = -\frac{\mathbf{a}}{\lvert\mathbf{a}\rvert}\, a\sinh(\mathbf{a})$
Arc tangent	$a\tan(\mathbf{a}) = -\frac{\mathbf{a}}{2\lvert\mathbf{a}\rvert}\log\{\frac{1+\mathbf{a}(\mathbf{a}/\lvert\mathbf{a}\rvert)}{1-\mathbf{a}(\mathbf{a}/\lvert\mathbf{a}\rvert)}\}$
Hyperbolic cosine	$\cosh(\mathbf{a}) = (\exp(\mathbf{a}) + 1/\exp(\mathbf{a}))/2$
Hyperbolic sine	$\sinh(\mathbf{a}) = (\exp(\mathbf{a}) - 1/\exp(\mathbf{a}))/2$
Hyperbolic tangent	$\tanh(\mathbf{a}) = \frac{\sinh(\mathbf{a})}{\cosh(\mathbf{a})}$
Arc hyperbolic cosine	$a\cosh(\mathbf{a}) = \log(\mathbf{a} + \sqrt{\mathbf{a}^2 - 1})$
Arc hyperbolic sine	$a\sinh(\mathbf{a}) = \log(\mathbf{a} + \sqrt{\mathbf{a}^2 + 1})$
Arc hyperbolic tangent	$a\tanh(\mathbf{a}) = \frac{1}{2}\log(\frac{1+\mathbf{a}}{1-\mathbf{a}})$
Exponential function	$\exp(\mathbf{a}) = \exp(a_0 + \lvert\mathbf{a}\rvert\frac{\mathbf{a}}{\lvert\mathbf{a}\rvert}) = \exp(a_0)\exp(\lvert\mathbf{a}\rvert\frac{\mathbf{a}}{\lvert\mathbf{a}\rvert}) = \exp(a_0)(\cos(\lvert\mathbf{a}\rvert), \sin(\lvert\mathbf{a}\rvert)\frac{\mathbf{a}}{\lvert\mathbf{a}\rvert})$
Quaternion raised to a power	$\mathbf{a}^n = \exp(n\,\ell n(\mathbf{a})) = \exp(n\,\ell n(\lvert\mathbf{a}\rvert) + na\tan(\frac{\lvert\mathbf{a}\rvert}{a_0})\frac{\mathbf{a}}{\lvert a\rvert}) = \lvert\mathbf{a}\rvert^n(\cos(na\tan(\frac{\lvert\mathbf{a}\rvert}{a_0}), \sin(na\tan(\frac{\lvert\mathbf{a}\rvert}{a_0}))\frac{\mathbf{a}}{\lvert a\rvert})$

III. Natural Logarithm: $\ell n(x)$

$$
\begin{aligned}
\ell n(\mathbf{a}) &= \ell n(a_0 + \mathbf{a}) = \ell n(a_0) + \ell n\left(1 + \frac{\mathbf{a}}{a_0}\right) \\
&= \ell n(a_0) + \left(\frac{\mathbf{a}}{a_0} - \frac{1}{2}\frac{\mathbf{a}}{a_0}\frac{\mathbf{a}}{a_0} + \frac{1}{3}\frac{\mathbf{a}}{a_0}\frac{\mathbf{a}}{a_0}\frac{\mathbf{a}}{a_0} - \frac{1}{4}\frac{\mathbf{a}}{a_0}\frac{\mathbf{a}}{a_0}\frac{\mathbf{a}}{a_0}\frac{\mathbf{a}}{a_0} + \cdots\right) \\
&= \ell n(a_0) + \left(\frac{\mathbf{a}}{a_0} + \frac{1}{2}\frac{\lvert\mathbf{a}\rvert^2}{a_0^2} - \frac{1}{3}\frac{\lvert\mathbf{a}\rvert^3}{a_0^3}\frac{\mathbf{a}}{\lvert\mathbf{a}\rvert} - \frac{1}{4}\frac{\lvert\mathbf{a}\rvert^4}{a_0^4} + \cdots\right) \\
&= \ell n(a_0) + \left(\frac{1}{2}\frac{\lvert\mathbf{a}\rvert^2}{a_0^2} - \frac{1}{4}\frac{\lvert\mathbf{a}\rvert^4}{a_0^4} + \cdots\right) \\
&\quad + \left(\frac{\lvert\mathbf{a}\rvert}{a_0} - \frac{1}{3}\frac{\lvert\mathbf{a}\rvert^3}{a_0^3} + \cdots\right)\frac{\mathbf{a}}{\lvert\mathbf{a}\rvert} = \left(\ell n(\lvert\mathbf{a}\rvert), a\tan\left(\frac{\lvert\mathbf{a}\rvert}{a_0}\right)\frac{\mathbf{a}}{\lvert\mathbf{a}\rvert}\right)
\end{aligned}
$$

Appendix B: Derivation for Alternate Forms for Trig Functions

Quaternion algebra permits several alternative forms to exist for elementary functions. To this end, a basic derivation is presented to establish the equality for the following alternative forms for the quaternion sine function:

$$
\sin(\mathbf{q}) = \sin(q_0 + \mathbf{q}) = \frac{(\exp(\hat{u}\mathbf{q}) - \exp(-\hat{u}\mathbf{q}))}{2\hat{u}}; \qquad \hat{u} = \frac{\mathbf{q}}{\lvert\mathbf{q}\rvert}
$$

From Appendix A, it follows that the expression for the first term is given by

$$
\begin{aligned}
\sin(\mathbf{q}) &= \sin(q_0 + \mathbf{q}) = \sin(q_0)\cos(\mathbf{q}) + \cos(q_0)\sin(\mathbf{q}) \\
&= \left(\sin(q_0)\cosh(\lvert\mathbf{q}\rvert), \cos(q_0)\sinh(\lvert\mathbf{q}\rvert)\frac{\mathbf{q}}{\lvert\mathbf{q}\rvert}\right)
\end{aligned}
$$

where Appendix A equations have been used to simplify the vector calculations for sin(*) and cos(*).

The second term corresponds to the classical complex-variable description for the sine function. This equation is establish through a series of transformations. First, the product of the unit vector and the quaternion is simplified as follows:

$$
\begin{aligned}
\hat{u}\mathbf{q} &= \hat{u}(q_0 + \mathbf{q}) = (0 - \hat{u}\cdot\mathbf{q}, 0 + q_0\hat{u} + \hat{u}x\mathbf{q}) = (-\lvert\mathbf{q}\rvert, q_0\hat{u}) \\
-\hat{u}\mathbf{q} &= -\hat{u}(q_0 + \mathbf{q}) = (0 + \hat{u}\cdot\mathbf{q}, 0 - q_0\hat{u} - \hat{u}x\mathbf{q}) = (\lvert\mathbf{q}\rvert, -q_0\hat{u})
\end{aligned}
$$

Next, computing the quaternion exponential terms, one obtains

$$
\begin{aligned}
\exp(\hat{u}\mathbf{q}) &= \exp(-\lvert\mathbf{q}\rvert, q_0\hat{u}) = \exp(-\lvert\mathbf{q}\rvert)\exp(q_0\hat{u}) \\
&= \exp(-\lvert\mathbf{q}\rvert)(\cos(q_0), \sin(q_0)\hat{u}) \\
\exp(-\hat{u}\mathbf{q}) &= \exp(\lvert\mathbf{q}\rvert, -q_0\hat{u}) = \exp(\lvert\mathbf{q}\rvert)\exp(-q_0\hat{u}) \\
&= \exp(\lvert\mathbf{q}\rvert)(\cos(-q_0), \sin(-q_0)\hat{u}) \\
&= \exp(\lvert\mathbf{q}\rvert)(\cos(q_0), -\sin(q_0)\hat{u})
\end{aligned}
$$

leading to the exponential difference calculation:

$$
\begin{aligned}
\exp(\hat{u}\mathbf{q}) - \exp(-\hat{u}\mathbf{q}) &= (\{\exp(-\lvert\mathbf{q}\rvert) \\
&\quad - \exp(\lvert\mathbf{q}\rvert)\}\cos(q_0), \{\exp(-\lvert\mathbf{q}\rvert) + \exp(\lvert\mathbf{q}\rvert)\}\sin(q_0)\hat{u}) \\
&= 2(-\sinh(\lvert\mathbf{q}\rvert)\cos(q_0), \cosh(\lvert\mathbf{q}\rvert)\sin(q_0)\hat{u})
\end{aligned}
$$

Table B1 Special-case solutions for quaternion linear equations

Case	Coefficient assumptions	Necessary conditions	Solution	Comments
1	$\mathbf{a} = \mathbf{b}, \mathbf{a}_0 \neq \mathbf{0}, \mathbf{c} \neq \mathbf{0}$	$\begin{pmatrix} 2\mathbf{a}_0\mathbf{x}_0 - 2\mathbf{a}\cdot\mathbf{x} + \mathbf{c}_0 \\ 2\mathbf{a}_0\mathbf{x} + 2\mathbf{a}\mathbf{x}_0 + \mathbf{c} \end{pmatrix} = \begin{pmatrix} 0 \\ \mathbf{0} \end{pmatrix}$	$\mathbf{x}_0 = -\frac{\mathbf{c}_0\mathbf{a}_0 + \mathbf{a}\cdot\mathbf{c}}{2(\mathbf{a}_0^2 + \mathbf{a}\cdot\mathbf{a})}\ \mathbf{x} = \frac{-(2\mathbf{a}\mathbf{x}_0 + \mathbf{c})}{2\mathbf{a}_0}$	Well-behaved solution
2	$\mathbf{a} = \mathbf{b}, \mathbf{a}_0 \neq \mathbf{0}, \mathbf{a} = -\mathbf{c}$	$\begin{pmatrix} -2\mathbf{a}\cdot\mathbf{x} - \mathbf{c}_0 \\ 2\mathbf{x}_0\mathbf{a} + \mathbf{c} \end{pmatrix} = \begin{pmatrix} 0 \\ \mathbf{0} \end{pmatrix}$	No solution exists unless the following constraints are satisfied: $\mathbf{x}_0 = \frac{\mathbf{c}\cdot\mathbf{x}}{\mathbf{c}_0}$ and $\mathbf{a} = \frac{-\mathbf{c}}{2\mathbf{x}_0}$, for example. Let $\mathbf{x}_0 = \frac{1}{2}$ and $\mathbf{a} = -\mathbf{c} \Rightarrow \mathbf{x}\left(1 - \frac{\mathbf{c}_0\mathbf{c}}{\mathbf{c}\cdot\mathbf{c}}\right)\Big/2$	Minimum norm solution for special case
3	$\mathbf{a} = \mathbf{b}, \mathbf{a}_0 = \mathbf{0}, \mathbf{c} = \mathbf{0}$	$\begin{pmatrix} -2\mathbf{a}\cdot\mathbf{x} \\ 2\mathbf{x}_0\mathbf{a} \end{pmatrix} = \begin{pmatrix} 0 \\ \mathbf{0} \end{pmatrix}$	$x_0 = 0\ \mathbf{x} = \mathbf{b}$, where $\mathbf{b}\cdot\mathbf{a} = 0$ if $\mathbf{b} = \mathbf{0}$ then any $\mathbf{x}\cdot\mathbf{a} = 0$ is a solution	
4	$\mathbf{a} = \mathbf{b}, \mathbf{a}_0 \neq \mathbf{0}, \mathbf{c} = \mathbf{0}$	$\begin{pmatrix} 2\mathbf{a}_0\mathbf{x}_0 - 2\mathbf{a}\cdot\mathbf{x} \\ 2\mathbf{a}_0\mathbf{x} + 2\mathbf{a}\mathbf{x}_0 \end{pmatrix} = \begin{pmatrix} 0 \\ \mathbf{0} \end{pmatrix}$	$\mathbf{x}_1 = \hat{\imath} - \mathbf{a}_1\hat{\mathbf{k}}/\mathbf{a}_3$ new solution $\mathbf{x}_2 = \hat{\mathbf{j}} - \mathbf{a}_2\hat{\mathbf{k}}/\mathbf{a}_3$ Meister [13,14]	Null space solution double root at $2\mathbf{a}_0$
5	$\mathbf{b} = -\mathbf{a}, \mathbf{c} = \mathbf{0}$	$\widetilde{2\mathbf{a}}\cdot\mathbf{x} = \mathbf{0}$	$\mathbf{x} = \mathbf{x}_0 + b\mathbf{a}$ Meister [13,14]	$\mathbf{x}_0$and b arbitrary constants
6	$\mathbf{b} = -\mathbf{a}, \mathbf{c}_0 = \mathbf{0}, \mathbf{c} \neq 0$	$\widetilde{2\mathbf{a}}\cdot\mathbf{x} + \mathbf{c} = \mathbf{0}$	If $\mathbf{a}\cdot\mathbf{c} = 0$, then every $\mathbf{x}\cdot\mathbf{c} = 0$, satisfying $\widetilde{2\mathbf{a}}\cdot\mathbf{x} + \mathbf{c} = 0$ is a solution	Special-case solutions

Table B2 Numerical test cases

Equation	Coefficients	Quaternion solution		Solution Accuracy
Linear	$\mathbf{a}=\mathbf{b}$, $\mathbf{a}_0 \neq \mathbf{0}$, & $\mathbf{c}\neq\mathbf{0}$, $\mathbf{a}\neq\mathbf{b}$, $\mathbf{c}\neq\mathbf{0}$	$\mathbf{x} = -0.6958 - 1.7320\hat{\imath} - 0.8660\hat{\mathbf{j}} - 2.0784\hat{\mathbf{k}}$		$\sim 1.0\times 10^{-15}$
Linear	$\mathbf{a}\neq\mathbf{b}$, $\mathbf{c}\neq\mathbf{0}$	$\mathbf{x} = \frac{3}{2} + \frac{1}{2}\hat{\imath} + \frac{1}{2}\hat{\mathbf{j}} - \frac{3}{2}\hat{\mathbf{k}}$		Exact solution
Quadratic	$\mathbf{a}\neq\mathbf{b}$, $\mathbf{c}\neq\mathbf{0}$	$p^+ = 1.9158 + 0.7471\hat{\imath} - 5.2526\hat{\mathbf{j}} + 3.1470\hat{\mathbf{k}}$	$p^- = -0.7611 - 0.7471\hat{\imath} - 0.5208\hat{\mathbf{j}} - 0.8376\hat{\mathbf{k}}$	$\sim 1.0\times 10^{-16}$
Quadratic	$\mathbf{a}\neq\mathbf{b}$, $\mathbf{c}\neq\mathbf{0}$	$\mathbf{p}_1 = 0.4676 - 1.9651\hat{\imath} - 0.9059\hat{\mathbf{j}} - 0.4992\hat{\mathbf{k}}$	$\mathbf{p}_2 = 0.5323 + 0.8911\hat{\imath} - 0.3165\hat{\mathbf{j}} - 1.0935\hat{\mathbf{k}}$	$\sim 1.0\times 10^{-12}$

This expression is multiplied from the left into the difference expression for quaternion exponentials (i.e., the order of the multiplication is not important, because the same vector part of the quaternion is used in all operations, which leads to a $\hat{u}\times\hat{u}=0$ in the corresponding quaternion products). One can now recognize the hyperbolic expressions, leading to

$$\begin{aligned}&\frac{(\exp(\hat{u}\mathbf{q}) - \exp(-\hat{u}\mathbf{q}))}{2\hat{u}}\\ &= \frac{1}{\hat{u}}(-\sinh(|\mathbf{q}|)\cos(q_0), \cosh(|\mathbf{q}|)\sin(q_0)\hat{u})\\ &= (\cosh(|\mathbf{q}|)\sin(q_0), \sinh(|\mathbf{q}|)\cos(q_0)\hat{u})\end{aligned}$$

which agrees with A.6, thereby establishing the validity of the alternate form for the quaternion sine.

References

[1] Gibbon, J. D., Holm, D. D., Kerr, R. M., and Roulstone, I., "Quaternions and Particle Dynamics in the Euler Fluid Equations," *Nonlinearity*, Vol. 19, No. 8, Aug. 2006, pp. 1969–1983. doi:10.1088/0951-7715/19/8/011

[2] Adler, S. L., *Quaternionic Quantum Mechanics and Quantum Fields*, Oxford Univ. Press, New York, 1005.

[3] Jiang, T., "An Algorithm for Eigenvalues and Eigenvectors of Quaternion Matrices in Quaternionic Quantum Mechanics," *Journal of Mathematical Physics (Woodbury, New York)*, Vol. 45, No. 8, Aug. 2004, pp. 3334–3338. doi:10.1063/1.1769106

[4] Turner, J. D., "Solving Linear and Quadratic Quaternion Equations," *Journal of Guidance, Control, and Dynamics*, Vol. 29, No. 6, Nov.–Dec. 2006, pp. 1420–1423. doi:10.2514/1.21488

[5] Junkins, J. L., and Turner, J. D., *Optimal Spacecraft Rotational Maneuvers: Studies in Astronautics*, Vol. 3, Elsevier, Amsterdam, 1986.

[6] Schaub, H., and Junkins, J. L., *Analytical Mechanics of Space Systems*, AIAA Education Series, AIAA, Reston, VA, 2003.

[7] Meister, L., "Quaternions and their Application in Photogrammetry and Navigation," *Habilitationsschrift of Ljudmila Meister*, Technische Univ. Bergakademie, Freiberg, Germany, 1997.

[8] Meister, L., and Schaeben, H., "A Concise Quaternion Geometry of Rotations," *Mathematical Methods in the Applied Sciences*, Vol. 27, 2004.

[9] Hamilton, W. R., *Elements of Quaternions*, Longmans Green, London, 1866.

[10] Joly, C. J., *A Manual of Quaternions*, Macmillan, London, 1905.

[11] Guerlebeck, K., and Sproessig, W., *Quaternionic and Clifford Calculus for Physicists and Engineers*, Wiley, New York, 1998.

[12] Sorensen, D. C., and Zhou, Y., "Direct Methods for Matrix Sylvester and Lyapunov Equations," *Journal of Applied Mathematics*, Vol. 2003, No. 6, 2003, pp 277–303. doi:10.1155/S1110757X03212055

[13] Zhang, F., "Quaternions and Matrices of Quaternions," *Linear Algebra and Its Applications*, Vol. 251, 1997, pp. 21–57. doi:10.1016/0024-3795(95)00543-9

[14] Brenner, J. L., "Matrices of Quaternions," *Pacific Journal of Mathematics*, Vol. 1, 1951, pp. 329–335.

[15] Gerstner, A. B., Byers, R., and Mehrmann, V., "A Quaternion QR Algorithm," *Numerische Mathematik*, Vol. 55, 1989, pp. 83–95. doi:10.1007/BF01395873

[16] Lee, H. C., "Eigenvalues of Canonical Forms of Matrices with Quaternion Coefficients," *Proceedings of the Royal Irish Academy, Section A (Mathematical and Physical Sciences)*, Vol. 52, 1949.

[17] Niven, I., "Equations in Quaternions," *The American Mathematical Monthly*, Vol. 48, No. 10, 1941, pp. 654–661. doi:10.2307/2303304

[18] Costa, C., and Serodio, R., "A Footnote on Quaternion Block-Tridiagonal Systems," *Electronic Transactions on Numerical Analysis*, Vol. 9, 1999, pp. 53–55.

[19] Tu, C. J., "Cramer Rules for Weighted System over the Quaternion Field," *Journal of the Zhangzhou Teachers College (Natural Science Editions)*, Vol. 9, No. 2, 1995, pp. 17–22.

[20] Banachiewicz, T., "Zur Berechungung der Determinaten, ie auch der Inversen und zur Darauf Basierten Auflösung der Systeme Linearer Gleichungen," *Acta Astronomica. Série C.*, Vol. 3, 1937, pp. 41–67.