THE APPLICATION OF QUATERNION ALGEBRA TO GYROSCOPIC MOTION, NAVIGATION, AND GUIDANCE

George P. Davailus

Old Dominion University
Aerospace Engineering Department

Advisor: Brett A. Newman

Abstract

Currently, many six degree of freedom (6-DOF) trajectory simulations and simulations of gyroscopic motion use quaternions to define a vehicle's orientation. Of those that do, however, none take full advantage of the properties of quaternion algebra. Quaternions are also known as hypercomplex numbers. They can be treated as individual quantities for which all the standard algebraic operations are defined. Consequently, they have advantages that Euler angles and transformation matrices do not.

This paper will describe the use of quaternion algebra and elliptic functions to obtain a closed form solution for torque free gyroscopic motion in terms of the rotational quaternion and its derivative. It will also define an alternative 6-DOF formulation, that, when combined with quaternion algebra, is potentially much more powerful than current simulations. Lastly, the paper will present a special case of spin stabilized rigid bodies.

Introduction

Review of Literature

Due to the obscure nature of the subject, the prior literature is sparse. In the 19th century, the quaternion proponents struggled for dominance with the vector proponents. Vectors won the war, and quaternions have been somewhat out of favor ever since. Elliptic functions constitute another obscure subject. However, they have been used for torque-free rigid body motion. Synge and Griffith derive a closed form solution for the angular rates and two of the three Euler angles.[2] Davailus developed a 6-DOF formulation that numerically integrated the first and second quaternion derivatives, thereby improving the conditioning of the differential equations.[1] Mathematica has a fairly extensive set of elliptic library functions; their documentation lists various properties and formulas involving elliptic functions.[5]

Definition of terms

q	rotational quaternion
$\dot{q}$	quaternion first derivative
$\ddot{q}$	quaternion second derivative
Q	matrix of quaternion components
$\vec{\omega}$	angular velocity vector
$\dot{\vec{\omega}}$	angular acceleration vector
I	mass moment of inertia matrix
I^{-1}	inverse mass moment of inertia
$\vec{M}$	moment vector

q_{ω} quaternion angular velocity
$q_{\dot{\omega}}$ quaternion angular acceleration
q_{λ} quaternion eigenvalue
q_{e} quaternion eigenvector
a scalar part of quaternion
b scalar part of quaternion
$\vec{u}$ vector part of quaternion
$\vec{w}$ vector part of quaternion

Background

The computation of trajectories that involve rigid body motion requires the numerical solution of a large set of ordinary differential equations.

$$x(0) = x_0$$
$$\frac{dx}{dt} = f(x,t)$$

For typical 6-DOF simulations, the components of the unknown vector *x* include the three components of angular velocity vector – roll, pitch, and yaw – and the four components of the rotational quaternion. The quaternion can represent a transformation of a vector from one frame of reference or coordinate system to another, often from the rigid body frame to an inertial frame or a reference frame, or vice-versa. Depending on the trajectory, all of these components can oscillate rapidly. The relevant state equations are as follows. [3, 4]

$$\dot{q} = \frac{1}{2} Q \vec{\omega} \qquad (1)$$
$$\dot{\vec{\omega}} = I^{-1}(\vec{M} - \vec{\omega} \times I \vec{\omega}) \qquad (2)$$

Note that the quaternion is not a vector. Since these quantities are changing continuously, one must usually numerically integrate them to determine their value at a particular point in time. By changing the differential equations so that they are better conditioned, one can increase the integration step size. This paper will describe two approaches one can utilize to accomplish this. One is for general rigid body motion, and the other applies only to spin stabilized rigid bodies.

Alternative 6-DOF Formulations

This section describes two methods that can be used to improve the conditioning of the differential equations so that larger integration step sizes can be used. The first subsection explains the removal of the angular velocity from the state vector and its replacement by the quaternion first derivative. The second subsection describes fitting a variable amplitude and frequency sine wave to the quaternion components (i.e., the amplitudes and phase angles become state variables) for the case of spin stabilized bodies. This paper will only concern itself with two frames of reference: a rigid body frame with an origin at the rigid body's center of mass, and an inertial frame.

Quaternion Second Derivative

Numerical integration of highly oscillatory state variables generally requires eight to ten integration cycles per cycle of the state variable. Therefore changing the differential equations in such a way that the frequency of the state variables is reduced allows one to increase the numerical integration step size. Often the angular velocity vector is less tractable than the rotational quaternion, so it is more of an obstacle to higher step sizes than the quaternion is. The solution is to replace the angular velocity with the rotational quaternion derivative. Define the matrix *Q*.

$$Q = \begin{bmatrix} -q_2 & -q_3 & -q_4 \\ q_1 & -q_4 & q_3 \\ q_4 & q_1 & -q_2 \\ -q_3 & q_2 & q_1 \end{bmatrix} \quad (3)$$

The quaternion first and second derivatives are given as follows.

$$\dot{q} = \frac{1}{2} Q \vec{\omega} \quad (4)$$

$$\ddot{q} = \frac{1}{2} \dot{Q} \vec{\omega} + \frac{1}{2} Q \dot{\vec{\omega}} \quad (5)$$

There is only one problem remaining: computing the updated angular velocity. But this is easily solved. Because the columns of *Q* are orthogonal to each other, one can simply multiply the transpose of Q by the updated quaternion derivative.

$$\vec{\omega}_{i+1} = 2 Q_{i+1}^T \dot{q}_{i+1} \quad (6)$$

Written term by term one gets the following.

$$\omega_1 = 2\left(-q_2 \dot{q}_1 + q_1 \dot{q}_2 + q_4 \dot{q}_3 - q_3 \dot{q}_4\right) \quad (7)$$

$$\omega_2 = 2\left(-q_3 \dot{q}_1 - q_4 \dot{q}_2 + q_1 \dot{q}_3 + q_2 \dot{q}_4\right) \quad (8)$$

$$\omega_3 = 2\left(-q_4 \dot{q}_1 + q_3 \dot{q}_2 - q_2 \dot{q}_3 + q_1 \dot{q}_4\right) \quad (9)$$

The key point to keep in mind here is that although this approach does not allow avoiding the computation of the angular velocity or acceleration, it allows the model to avoid numerically integrating the angular acceleration to update the angular velocity. A comparison between the old 6-DOF model and the new one is given below.[1]

Initial Conditions

Old 6-DOF Model	New 6-DOF Model
q_0	q_0
$\vec{\omega}_0$	$\vec{\omega}_0$
	$\dot{q}_0 = \frac{1}{2} Q \vec{\omega}$

Derivative Computations

Old 6-DOF Model	New 6-DOF Model
$\dot{q} = \frac{1}{2} Q \vec{\omega}$	$\vec{\omega} = 2 Q^T \dot{q}$
$\dot{\vec{\omega}} = I^{-1}(\vec{M} - \vec{\omega} \times I \vec{\omega})$	$\dot{\vec{\omega}} = I^{-1}(\vec{M} - \vec{\omega} \times I \vec{\omega})$
	$\ddot{q} = \frac{1}{2}(\dot{Q} \vec{\omega} + Q \dot{\vec{\omega}})$

Numerical Integration

Old 6-DOF Model	New 6-DOF Model
Numerically integrate $\dot{q}$ to update q	Numerically integrate $\dot{q}$ to update q
Numerically integrate $\dot{\omega}$ to update ω	Numerically integrate $\ddot{q}$ to update $\dot{q}$

Often the new differential equations have better conditioning and therefore larger integration step sizes can be taken with them.[1]

Quaternion Algebra

As stated earlier, quaternions can be viewed as individual quantities for which all the standard algebraic operations are defined. The first subsection of this section will describe the algebraic operations. The second subsection will discuss potential new navigation, guidance, and control algorithms that take advantage of the properties of quaternion algebra. The third subsection describes integrating elliptic functions with quaternion algebra to obtain a closed form solution to the problem of torque free gyroscopic motion.

Quaternion Operations

First define a quaternion as a quantity with a real scalar and an imaginary three dimensional vector.

$$q = \begin{bmatrix} a \\ \vec{u} \end{bmatrix} \tag{10}$$

Addition

$$\begin{bmatrix} a \\ \vec{u} \end{bmatrix} + \begin{bmatrix} b \\ \vec{w} \end{bmatrix} = \begin{bmatrix} a+b \\ \vec{u}+\vec{w} \end{bmatrix} \tag{11}$$

Subtraction

$$\begin{bmatrix} a \\ \vec{u} \end{bmatrix} - \begin{bmatrix} b \\ \vec{w} \end{bmatrix} = \begin{bmatrix} a-b \\ \vec{u}-\vec{w} \end{bmatrix} \tag{12}$$

Additive Identity

$$\begin{bmatrix} 0 \\ \vec{0} \end{bmatrix} \tag{13}$$

Multiplication

$$\begin{bmatrix} a \\ \vec{u} \end{bmatrix} \times \begin{bmatrix} b \\ \vec{w} \end{bmatrix} = \begin{bmatrix} ab + \vec{u}\cdot\vec{w} \\ a\vec{w} + b\vec{u} + i\vec{u}\times\vec{w} \end{bmatrix}, i = \sqrt{-1} \tag{14}$$

Note that quaternion multiplication is not commutative.

Multiplicative Identity

$$q_I = \begin{bmatrix} 1 \\ \vec{0} \end{bmatrix} \tag{15}$$

Vector Conjugate

$$q^* = \begin{bmatrix} a \\ \vec{u} \end{bmatrix}^* = \begin{bmatrix} a \\ -\vec{u} \end{bmatrix} \tag{16}$$

Determinant and Magnitude

$$\det(q) = |q|^2 = ab - \vec{u}\cdot\vec{w} \tag{17}$$

Inverse

$$q^{-1} = \frac{q^*}{\det(q)} \tag{18}$$

Division

$$q \div p = \frac{q \times p^*}{\det(p)} \tag{19}$$

Exponentials

$$\begin{aligned} e^q &= e^a(\cosh r + \hat{u}\sinh r) = e^a(\cos s + \hat{u}\sin s) \\ r &= is = |\vec{u}|_2 \end{aligned} \tag{20}$$

The above formulas lead to an analytical solution for constant angular velocity.

$$q(t+\Delta t) = q(t) \times e^{\frac{q_\omega \Delta t}{2}} \tag{21}$$

$$q_\omega = \begin{bmatrix} 0 \\ \vec{\omega} \end{bmatrix} \tag{22}$$

Here the angular velocity is being treated as a purely imaginary quaternion. The same can be done for the angular acceleration.

Square Roots

$$\sqrt{q} = \begin{pmatrix} \cos\frac{\delta}{2} \\ \cos\alpha\sin\frac{\delta}{2} \\ \cos\beta\sin\frac{\delta}{2} \\ \cos\gamma\sin\frac{\delta}{2} \end{pmatrix}^{1/2} = \begin{pmatrix} \cos\frac{\delta}{4} \\ \cos\alpha\sin\frac{\delta}{4} \\ \cos\beta\sin\frac{\delta}{4} \\ \cos\gamma\sin\frac{\delta}{4} \end{pmatrix} \tag{23}$$

Proof:

$$\sqrt{q}\times\sqrt{q} = \begin{bmatrix} \left(\cos\frac{\delta}{4}\right)^2 - \left(\sin\frac{\delta}{4}\right)^2\left[\cos^2\alpha+\cos^2\beta+\cos^2\gamma\right] \\ 2\cos\frac{\delta}{4}\sin\frac{\delta}{4}\begin{bmatrix} \cos\alpha \\ \cos\beta \\ \cos\gamma \end{bmatrix} \end{bmatrix} = \begin{pmatrix} \cos\frac{\delta}{2} \\ \cos\alpha\sin\frac{\delta}{2} \\ \cos\beta\sin\frac{\delta}{2} \\ \cos\gamma\sin\frac{\delta}{2} \end{pmatrix} \tag{24}$$

For non unit magnitude quaternions, normalize and take the square root; then multiply by the square root of the magnitude.

$$q^{\frac{1}{2}} = |q|_2^{\frac{1}{2}}\,\hat{q}^{\frac{1}{2}}$$

Quaternion Navigation and Guidance

The operations in the previous section allow one to combine the kinematic and kinetic rotational equations into one equation.

$$\ddot{q} - \frac{1}{2}\dot{q}\times q_\omega - \frac{1}{2}q\times q_{\dot{\omega}} = 0 \tag{25}$$

Written term by term one gets the following.

$$\ddot{q}_1 = \frac{1}{2}\left(-\dot{q}_2\omega_1 - \dot{q}_3\omega_2 - \dot{q}_4\omega_3 - q_2\dot{\omega}_1 - q_3\dot{\omega}_2 - q_4\dot{\omega}_3\right) \tag{26}$$

$$\ddot{q}_2 = \frac{1}{2}\left(\dot{q}_1\omega_1 + \dot{q}_3\omega_3 - \dot{q}_4\omega_2 + q_1\dot{\omega}_1 + q_3\dot{\omega}_3 - q_4\dot{\omega}_2\right) \tag{27}$$

$$\ddot{q}_3 = \frac{1}{2}\left(\dot{q}_1\omega_2 - \dot{q}_2\omega_3 + \dot{q}_4\omega_1 + q_1\dot{\omega}_2 - q_2\dot{\omega}_3 + q_4\dot{\omega}_1\right) \tag{28}$$

$$\ddot{q}_4 = \frac{1}{2}\left(\dot{q}_1\omega_3 + \dot{q}_2\omega_2 - \dot{q}_3\omega_1 + q_1\dot{\omega}_3 + q_2\dot{\omega}_2 - q_3\dot{\omega}_1\right) \tag{29}$$

The equation above can also be written as the following.

$$\ddot{q} - q\times\left(\frac{1}{2}q_{\dot{\omega}} - \frac{1}{4}q_\omega^2\right) = 0 \tag{30}$$

Written term by term one gets the following.

$$\ddot{q}_1 = \frac{1}{2}\left(\frac{1}{2}q_1|\vec{\omega}|_2^2 - q_2\dot{\omega}_1 - q_3\dot{\omega}_2 - q_4\dot{\omega}_3\right) \tag{31}$$

$$\ddot{q}_2 = \frac{1}{2}\left(\frac{1}{2}q_2|\vec{\omega}|_2^2 + q_1\dot{\omega}_1 + q_3\dot{\omega}_3 - q_4\dot{\omega}_2\right) \tag{32}$$

$$\ddot{q}_3 = \frac{1}{2}\left(\frac{1}{2}q_3|\vec{\omega}|_2^2 + q_1\dot{\omega}_2 - q_2\dot{\omega}_3 + q_4\dot{\omega}_1\right) \tag{33}$$

$$\ddot{q}_4 = \frac{1}{2}\left(\frac{1}{2} q_4 |\vec{\omega}|_2^2 + q_1\dot{\omega}_3 + q_2\dot{\omega}_2 - q_3\dot{\omega}_1\right) \quad (34)$$

Each term in the above equation is a quaternion. This allows one to compute the eigenvalues and eigenvectors using the quaternion algebra version of the quadratic formula.

$$\begin{bmatrix} \dot{q} \\ \ddot{q} \end{bmatrix} = \begin{bmatrix} 0 & q_I \\ \frac{q_{\dot{\omega}}}{2} & \frac{q_{\omega}}{2} \end{bmatrix} \begin{bmatrix} q \\ \dot{q} \end{bmatrix} \quad (35)$$

$$\det(I\lambda - A) = q_\lambda^2 + bq_\lambda + cq_I \quad (36)$$

$$q_\lambda = \frac{-\frac{q_\omega}{2} \pm \sqrt{\left(\frac{q_\omega}{2}\right)^2 - 2q_{\dot{\omega}} \times q_I}}{2q_I} \quad (37)$$

The eigenvector is given by the following.

$$q_e = \begin{bmatrix} q_I \\ q_\lambda \end{bmatrix} \quad (38)$$

This could potentially be developed into an eigenvalue feedback algorithm. This possibility will be further explored in the final draft.

Torque Free Gyroscopic Motion

If the angular velocity is changing continuously, the following holds true.

$$q(t_1) = q(t_0) \times e^{\int_{t_0}^{t_1} \frac{q_\omega}{2} dt} \quad (39)$$

The following is not exactly a proof, but one can take the following example with real numbers as an illustration.

$\dot{x} = ax$, where a is a constant.

Then the solution is given by the following.

$$x(t) = e^{at} \quad (40)$$

If a varies with time then the solution is given by the following.

$$x(t) = e^{\int a(t)dt} \quad (41)$$

In the case of the quaternion and the angular velocity, the equation above is simply being applied to hyper-complex numbers instead of real ones. Back substitution of the quaternion result into the governing differential equation verifies the validity of the closed form solution.

The closed form solutions for the roll, pitch, and yaw, respectively are given by the following. The elliptic functions *sn*, *cn*, and *dn* are used.[2]

$$\omega_1 = \alpha dn(p(t_1 - t_0), k) \quad (42)$$

$$\omega_2 = \beta sn(p(t_1 - t_0), k) \quad (43)$$

$$\omega_3 = \gamma cn(p(t_1 - t_0), k) \quad (44)$$

So the solution for the quaternion becomes the following.

$$q(t_1) = q(t_0) \times e^{\int_{t_0}^{t_1} \begin{bmatrix} 0 \\ \alpha dn(pt,k) \\ \beta sn(pt,k) \\ \gamma cn(pt,k) \end{bmatrix} dt} \quad (45)$$

The integrals for the elliptic functions are given by the following.[5]

$$\int sn(t,k)dt = \frac{1}{\sqrt{k}} \log(dn(t,k) - \sqrt{k} cn(t,k)) \quad (46)$$

$$\int cn(t,k)dt = \frac{\arccos(dn(t,k) sn(t,k))}{\sqrt{1 - dn^2(t,k)}} \quad (47)$$

$$\int sn(t,k)dt = am(t,k) \tag{48}$$

This leads to the following closed form solution to the torque free gyroscopic motion problem in terms of quaternions.

$$q(t_1) = q(t_0) \times e^{\begin{bmatrix} 0 \\ \alpha\left[am(t_1,k) - am(t_0,k)\right] \\ \beta\left[\frac{1}{\sqrt{k}}\log\left(dn(t_1,k) - \sqrt{k}\,cn(t_1,k)\right) - \frac{1}{\sqrt{k}}\log\left(dn(t_0,k) - \sqrt{k}\,cn(t_0,k)\right)\right] \\ \gamma\left[\frac{\arccos\left(dn(t_1,k)sn(t_1,k)\right)}{\sqrt{1 - dn^2(t_1,k)}} - \frac{\arccos\left(dn(t_0,k)sn(t_0,k)\right)}{\sqrt{1 - dn^2(t_0,k)}}\right] \end{bmatrix}}$$

Results

Experimental results will be presented in the final draft of a conference paper that will be presented by the author at the American Institute of Aeronautics and Astronautics this August in San Francisco.

References

[1] G. P. Davailus, “The Transformation of Oscillatory Equations in Six Degree of Freedom Trajectory Simulations Involving Coordinate Transformations”, M.S. Thesis, Department of Computer Science, Virginia Polytechnic Institute and State University, Blacksburg, Virginia, 1994.

[2] B. A. Griffith, J. L. Synge, “Principles of Mechanics”, McGraw-Hill Book Company, Inc., 1959, pps. 327-383.

[3] F. L. Lewis, B. L. Stevens, “Aircraft Control and Simulation”, John Wiley and Sons Inc., 2003, pps. 1-53.

[4] M. J. Sidi, “Spacecraft Dynamics and Control”, Cambridge University Press, 1997, pps. 88-111.

[5] Mathematica website, http://functions.wolfram.com/EllipticFunctions