Quaternion Computation

Neil Dantam
ntd@gatech.edu
Institute for Robotics and Intelligent Machines,
Georgia Institute of Technology, Atlanta, GA, USA

Work in Progress – Updated 2014-10-01.

***Abstract*—Quaternions are a useful representation for orientation, and dual quaternions extend the representation to handle translations as well. This report discusses computations that can be performed using quaternions. To accurately compute results near singularities, we provide Taylor series approximations which can be efficiently computed to within machine precision.**

CONTENTS

I. INTRODUCTION

Quaternions are a convenient representation for spatial motion that provides some computational advantages over other methods.

The straightforward definitions of many quaternion quantities, particularly exponentials, logarithms, and derivatives, contain singularities where a denominator goes to zero. We can avoid computational problems at these points by computing key factors near the singularity using a Taylor series, though this may require some careful rearrangement of terms to identify suitable factors and series.

A Taylor series evaluated near point a is:

$$
f(x) = \sum_{n=0}^{\infty} \frac{f^{(n)}(a)}{n!}(x-a)^n
= f(a) + \frac{f'(a)}{1!}(x-a) + \frac{f''(a)}{2!}(x-a)^2 + \frac{f^{(3)}(a)}{3!}(x-a)^3 + \dots \quad (1)
$$

To evaluate the infinite series to machine precision, we only need to compute up the term below floating point round-off.

The resulting approximation is a polynomial which can be efficiently evaluated using Horner's Rule, Algorithm 1. The coefficients are the terms $\frac{f(n)(a)}{n!}$ and the indeterminate variable is $x - a$. Note than many Taylor series have zero coefficents for the odd or even terms. We can produce a more compact Horner polynomial by omitting the zero coefficients, using $(x - a)^2$ as the indeterminate variable, and perhaps multiplying the whole result by $(x - a)$.

Algorithm 1: Horner's Rule

Input: $b_0, b_1, \dots b_n$: Coefficients
Input: z : Indeterminate Variable
Output: y : Result

$y \leftarrow b_n$
$y \leftarrow b_{n-1} + zy$
$y \leftarrow b_{n-2} + zy$
$\dots$
$y \leftarrow b_0 + zy$

A. Notation

We adopt the following abbreviations to condense notation:

- Quaternions are typeset as q.
- Dual Quaternions are typeset as $\mathcal{S}$.
- Vectors are typeset as $\vec{x}$.
- Matrices are typeset as $\mathbf{A}$.
- Time derivatives of variable x are given as $\dot{x}$.
- Sines and cosines are abbreviated as s and c.

II. Quaternions

Quaternions are an extension of the complex numbers, using basis elements i, j, and k defined as:

$$
i^2 = j^2 = k^2 = ijk = -1 \quad (2)
$$

From (2), it follows:

$$
jk = -kj = i \quad (3)
$$

$$
ki = -ik = j \quad (4)
$$

$$
ij = -ji = k \quad (5)
$$

A quaternion, then, is:

$$
q = w + xi + yj + zk \quad (6)
$$

Associative	$p \otimes (q \otimes r) = (p \otimes q) \otimes r$
Distributive	$p \otimes (q + r) = p \otimes q + p \otimes r$
NOT Commutative	$p \otimes q \neq q \otimes p$
Conjugate Mul.	$(p \otimes q)^* = q^* \otimes p^*$
Conjugate Add.	$(p + q)^* = q^* + p^*$

TABLE I
ALGEBRAIC QUATERNION PROPERTIES

A. Representation

We represent a quaternion as a 4-tuple of real numbers:

$$
\begin{aligned}
q &= w + xi + yj + zk \\
&= (x\ y\ z\ w) \\
&= \mathcal{H}(q_v,\ w)
\end{aligned} \quad (7)
$$

Historically, q_v is called the vector part of the quaternion and q_w the scalar part.

It is convenient to define quaternion operations in terms of vector and matrix operations, so we also the whole quaternion as a column vector. This also provides an in-memory storage representation.

$$
\vec{q} = [x\ y\ z\ w]^T \quad (8)
$$

$$
\vec{q_v} = [x\ y\ z]^T \quad (9)
$$

A alternate convention stores terms in $wxyz$ order, so when using different software packages, it is sometimes necessary to convert between orderings.

B. Multiplication

From the definition of the basis elements (2), we obtain a formula for quaternion multiplication. See section B for the detailed derivation.

1) Cross and dot product definition: We define quaternion multiplication in terms of cross products and dot products of its elements:

$$
q \otimes p = \begin{pmatrix} \vec{q_v} \times \vec{p_v} + q_w \vec{p_v} + p_w \vec{q_v} \\ q_w p_w - \vec{q_v} \cdot \vec{p_v} \end{pmatrix} \quad (10)
$$

2) Matrix definition: Expanding the above terms, we can express quaternion multiplication as matrix multiplication:

$$
\begin{aligned}
&q \otimes p = \\
\mathbf{Q_L}\vec{p} = &\begin{bmatrix} q_w & -q_z & q_y & q_x \\ q_z & q_w & -q_x & q_y \\ -q_y & q_x & q_w & q_z \\ -q_x & -q_y & -q_z & q_w \end{bmatrix} \vec{p} = \\
\mathbf{P_R}\vec{q} = &\begin{bmatrix} p_w & p_z & -p_y & p_x \\ -p_z & p_w & p_x & p_y \\ p_y & -p_x & p_w & p_z \\ -p_x & -p_y & -p_z & p_w \end{bmatrix} \vec{q} = \\
&\begin{bmatrix} q_x p_w + q_y p_z + q_w p_x - q_z p_y \\ q_z p_x + q_w p_y + q_y p_w - q_x p_z \\ q_w p_z + q_z p_w + q_x p_y - q_y p_x \\ -(q_y p_y + q_x p_x + q_z p_z - q_w p_w) \end{bmatrix}
\end{aligned} \quad (11)
$$

This matrix form is more suitable for efficient implementation computation using SIMD instructions.

3) Properties: Quaternion multiplication is associative and distributive, but it is not commutative.

4) Pure Multiplication: When multiplying by a pure quaternion, i.e., zero scalar part, we can simplify:

$$q \otimes (v, 0) = \begin{bmatrix} q_w & -q_z & q_y \\ q_z & q_w & -q_x \\ -q_y & q_x & q_w \\ -q_x & -q_y & -q_z \end{bmatrix} v = \begin{bmatrix} q_y v_z \\ q_z v_x \\ q_x v_y \\ -q_x v_x \end{bmatrix} - \begin{bmatrix} q_z v_y \\ q_x v_z \\ q_y v_x \\ q_y v_y \end{bmatrix} + \begin{bmatrix} q_w v_x \\ q_w v_y \\ q_w v_z \\ -q_z v_z \end{bmatrix} \tag{12}$$

$$(v, 0) \otimes q = \begin{bmatrix} q_w & q_z & -q_y \\ -q_z & q_w & q_x \\ q_y & -q_x & q_w \\ -q_x & -q_y & -q_z \end{bmatrix} v = \begin{bmatrix} v_y q_z \\ v_z q_x \\ v_x q_y \\ -v_x q_x \end{bmatrix} - \begin{bmatrix} v_z q_y \\ v_x q_z \\ v_y q_x \\ v_y q_y \end{bmatrix} + \begin{bmatrix} v_x q_w \\ v_y q_w \\ v_z q_w \\ -v_z q_z \end{bmatrix} \tag{13}$$

$$(u, 0) \otimes (v, 0) = \begin{bmatrix} u_y v_z - u_z v_y \\ u_z v_x - u_x v_z \\ u_x v_y - u_y v_x \\ -u_x v_x - u_y v_y - u_z v_z \end{bmatrix} = \begin{bmatrix} u \times v \\ -u \cdot v \end{bmatrix} \tag{14}$$

Thus, the case of multiplying two pure quaternions simplifies to the commonly used cross ($\times$) and dot ($\cdot$) products.

C. Norm

$$|q| = \sqrt{\vec{q} \cdot \vec{q}} \tag{15}$$

A unit quaternion has norm of one.

D. Conjugate

$$q^* = \mathcal{H}(-q_v,\ q_w) \tag{16}$$

E. Inverse

$$q^{-1} = \frac{q^*}{\vec{q} \cdot \vec{q}} \tag{17}$$

Note that for unit quaternions, the inverse is equal to the conjugate.

F. Exponential

The exponential shows the relationship between quaternions and complex numbers. Recall Euler's formula for complex numbers:

$$e^{i\theta} = \cos(\theta) + i \sin(\theta) \tag{18}$$

which relates the exponential function with angles in the complex plane. Similarly for quaternions, we can consider the angle between the real and imaginary parts, Figure 1, yielding

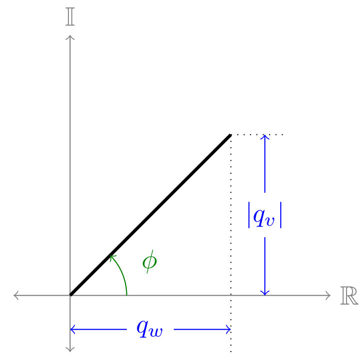

Fig. 1. Imaginary Plane for Quaternions

some useful trigonometric ratios for analyzing quaternion functions:

$$\phi = \operatorname{atan2}(|q_v|, q_w) \tag{19}$$

$$\sin(\phi) = \frac{|q_v|}{|q|} \tag{20}$$

$$\cos(\phi) = \frac{q_w}{|q|} \tag{21}$$

The quaternion exponential is:

$$e^q = e^{q_w} \mathcal{H}\left(q_v \frac{\sin(|q_v|)}{|q_v|},\ \cos(|q_v|) \right) \tag{22}$$

When $|q_v|$ approaches zero, we can use the Taylor series approximation:

$$\frac{\sin(\theta)}{\theta} = 1 - \frac{\theta^2}{6} + \frac{\theta^4}{120} - \frac{\theta^6}{5040} + \dots \tag{23}$$

For a pure quaternion, the exponential simplifies to:

$$q_w = 0 \implies \begin{cases} e^q = \mathcal{H}\left(q_v \frac{\sin(|q_v|)}{|q_v|},\ \cos(|q_v|) \right) \\ |e^q| = 1 \end{cases} \tag{24}$$

G. Logarithm

To compute the logarithm, first consider the angle between the vector and scalar parts of the quaternion.

$$\phi = \cos^{-1}\left(\frac{q_w}{|q|}\right) = \sin^{-1}\left(\frac{|q_v|}{|q|}\right) = \operatorname{atan2}(|q_v|, q_w) \tag{25}$$

The atan2 form to compute ϕ is generally best for numerical stability.

$$\ln q = \mathcal{H}\left(\frac{\phi}{|q_v|} q_v,\ \ln(|q|) \right) \tag{26}$$

When $|q_v|$ approaches zero, we can compute $\frac{\phi}{|q_v|}$ as follows:

$$\frac{\phi}{|q_v|} = \frac{\frac{\phi}{|q|}}{\frac{|q_v|}{|q|}} = \frac{\frac{\phi}{|q|}}{\sin(\phi)} = \frac{\frac{\phi}{\sin(\phi)}}{|q|} \tag{27}$$

Then, $\frac{\phi}{\sin(\phi)}$ can be approximated by Taylor series:

$$\frac{\theta}{\sin(\theta)} = 1 + \frac{\theta^2}{6} + \frac{7\theta^4}{360} + \frac{31\theta^6}{15120} + \dots \tag{28}$$

For a unit quaternion, the logarithm simplifies to:

$$\left|q\right| = 1 \implies \ln(q) = \mathcal{H}\left(\frac{\phi}{\sin(\phi)} q_v,\ 0\right) \tag{29}$$

H. Power

$$q^t = e^{t \ln q} \tag{30}$$

I. Pure Exponential Derivative

Becase quaternion multiplication is not commutative, the chain rule does not apply to the quaternion exponential derivative:

$$\frac{d \exp\left(f(q)\right)}{dt} \neq \frac{df(q)}{dt} \otimes \exp\left(f(q)\right) \tag{31}$$

The derivative of the exponential for a pure quaternion is:

$$\phi = |q_v| = \sqrt{q_v \cdot q_v} \tag{32}$$

$$\dot{\phi} = \frac{d\,|q_v|}{dt} = \frac{q_v \cdot \dot{q}_v}{\phi} \tag{33}$$

$$e^q = \mathcal{H}\left(\frac{\sin(\phi)}{\phi} q_v,\ \cos(\phi)\right) \tag{34}$$

$$\left(\frac{de^q}{dt}\right)_w = -\sin(\phi)\,\dot{\phi} = -\left(q_v \cdot \dot{q}_v\right)\frac{\sin(\phi)}{\phi} \tag{35}$$

$$\begin{aligned}\left(\frac{de^q}{dt}\right)_v &= \frac{s}{\phi}\dot{q}_v + \left(\frac{\dot{\phi}c}{\phi} - \frac{\dot{\phi}s}{\phi^2}\right) q_v = \\ &\frac{s}{\phi}\dot{q}_v + \left(\frac{c}{\phi^2} - \frac{s}{\phi^3}\right)\left(q_v \cdot \dot{q}_v\right) q_v = \\ &\frac{s}{\phi}\dot{q}_v + \left(\frac{c - \frac{s}{\phi}}{\phi^2}\right)\left(q_v \cdot \dot{q}_v\right) q_v\end{aligned} \tag{36}$$

Then, we handle the singularity for $\phi = 0$ using (23) and the following:

$$\frac{c}{\phi^2} - \frac{s}{\phi^3} = -\frac{1}{3} + \frac{\phi^2}{30} - \frac{\phi^4}{840} + \frac{\phi^6}{45360} + \dots \tag{37}$$

J. Unit Logarithm Derivative

The derivative of the unit quaternion logarithm is:

$$\begin{aligned}\ln q &= \frac{\phi}{\sin(\phi)} q_v \\ \frac{d \ln q}{dt} &= \frac{d\frac{\phi}{\sin(\phi)}}{dt} q_v + \frac{\phi}{\sin(\phi)}\dot{q}_v\end{aligned} \tag{38}$$

where we compute $\frac{d\frac{\phi}{\sin(\phi)}}{dt} q_v$ as follows:

$$\begin{aligned}\dot{\phi} &= \frac{d\cos^{-1}(w)}{dt} = \frac{\dot{w}}{\sin(\phi)} \\ \frac{d\frac{\phi}{\sin(\phi)}}{dt} &= \frac{\dot{\phi}}{\sin(\phi)} - \frac{\phi\dot{\phi}\cos(\phi)}{\sin^2(\phi)} \\ &= \dot{w}\left(-\frac{1}{\sin^2(\phi)} + \frac{\phi\cos(\phi)}{\sin^3(\phi)}\right)\end{aligned} \tag{39}$$

Representation	Storage
Quaternion	4
Axis-Angle	4
Rotation Vector	3
Euler Angles	3
Rotation Matrix	9

TABLE II
STORAGE REQUIREMENTS FOR ORIENTATION REPRESENTATIONS

Representation	Chain	Rotate Point
Quaternion	16 multiply, 12 add	15 multiply, 15 add
Rotation Matrix	27 multiply, 18 add	9 multiply, 6 add

TABLE III
COMPUTATIONAL REQUIREMENTS FOR ORIENTATION REPRESENTATIONS

Equations (38) and (39) have a singularity at $\phi = 0$. We can handle (38) with (28) and (39) with the following Taylor series:

$$-\frac{1}{\sin^2(\phi)} + \frac{\phi\cos(\phi)}{\sin^3(\phi)} = -\frac{1}{3} - \frac{2}{15}\phi^2 - \frac{2}{63}\phi^4 \dots \tag{40}$$

Alternatively, one could also use the Jacobian $\frac{\partial \ln q}{\partial q} =$

$$\begin{bmatrix} \frac{\phi}{\sin(\phi)} - \frac{\phi x^2}{\sin^3(\phi)} & -\frac{\phi xy}{\sin^3(\phi)} & -\frac{\phi xz}{\sin^3(\phi)} & -\frac{x}{\sin^2(\phi)} \\ -\frac{\phi xy}{\sin^3(\phi)} & \frac{\phi}{\sin(\phi)} - \frac{\phi y^2}{\sin^3(\phi)} & -\frac{\phi yz}{\sin^3(\phi)} & -\frac{y}{\sin^2(\phi)} \\ -\frac{\phi xz}{\sin^3(\phi)} & -\frac{\phi yz}{\sin^3(\phi)} & \frac{\phi}{\sin(\phi)} - \frac{\phi z^2}{\sin^3(\phi)} & -\frac{z}{\sin^2(\phi)} \end{bmatrix} \tag{41}$$

K. Unit Quaternion Angle

We can compute the angle between the vector forms of two unit quaternions as follows:

$$\begin{aligned}\angle(\vec{q}_1, \vec{q}_2) &= \cos^{-1}\left(\vec{q}_1 \cdot \vec{q}_2\right) = \\ &2\ \operatorname{atan2}\left(\left|q_1 - q_2\right|, \left|q_1 + q_2\right|\right)\end{aligned} \tag{42}$$

The atan2 form is more accurate [1].

L. Product Rule

Because quaternion multiplication is a linear operation (see section B), the product rule applies:

$$\frac{d}{dt}\left(q_1 \otimes q_2\right) = \dot{q}_1 \otimes q_2 + q_1 \otimes \dot{q}_2 \tag{43}$$

III. REPRESENTING ORIENTATION

A unit quaternion ($\left|q\right| = 1$) can represent an angular orientation.

A. Rotating a vector

We can rotate point v by unit quaternion q by computing $v' = \operatorname{rot}\left(q, v\right) = q \otimes v \otimes q^*$. Note that v is augmented with 0 in it's w position to perform the quaternion multiplication operation. Given this 0 value, the computation can be simplified to the following:

$$\begin{aligned}v' &= \operatorname{rot}\left(q, v\right) = \\ q \otimes v \otimes q^* &= 2\vec{q}_v \times \left(\vec{q}_v \times v + q_w v\right) + v\end{aligned} \tag{44}$$

which we can rewrite in a more SIMD-friendly form as:

$$\begin{aligned} a &= q_v \times v + q_w v \\ b &= q_v \times a \\ v' &= b + b + v \end{aligned} \tag{45}$$

B. Chaining rotations

Rotations q_1 and q_2 are chained by multiplying the two quaternions: $q_1 \otimes q_2$.

C. Angular Derivatives

Rotational velocity ω is related to the quaternion derivative as follows:

$$\dot{q} = \frac{1}{2}\omega \otimes q \tag{46}$$

$$\omega = 2\dot{q} \otimes q^* \tag{47}$$

Rotational acceleration $\dot{\omega}$ is related to the quaternion derivative as follows:

$$\ddot{q} = \frac{1}{2}\left(\dot{\omega} \otimes q + \omega \otimes \dot{q}\right) \tag{48}$$

$$= \frac{1}{2}\dot{\omega} \qquad = 2\left(\ddot{q} \otimes q^* + \dot{q} \otimes \dot{q}^*\right) \tag{49}$$

D. Axis-Angle

The axis-angle form, $a = (\hat{u}, \theta)$ represents rotation by angle θ around unit axis $\hat{u}$. We can also normalize the representation by scaling the axis by the angle $v = \theta\hat{u}$, which is sometimes called the rotation vector form.

Rotation vectors are related to unit quaternions through the exponential and logarithm.

$$\begin{aligned} q = \mathcal{H}\left(\hat{u}\sin\left(\frac{\theta}{2}\right),\ \cos\left(\frac{\theta}{2}\right)\right) = e^{\frac{\theta}{2}\hat{u}} = \\ \mathcal{H}\left(\frac{v}{|v|}\sin\left(\frac{|v|}{2}\right),\ \cos\left(\frac{|v|}{2}\right)\right) = e^{\frac{v}{2}} \end{aligned} \tag{50}$$

$$\theta = 2\cos^{-1}(q_w) = 2\tan^{-1}(|q_v|, q_w) = 2|\ln q| \tag{51}$$

$$\hat{u} = \begin{cases} \theta \neq 0 & \frac{q_v}{\sin\left(\frac{\theta}{2}\right)} \\ \theta = 0 & 0 \end{cases} = \frac{\ln q}{|\ln q|} \tag{52}$$

$$v = 2\ln q \tag{53}$$

The rotation vector and quaternion derivatives are related as follows, substituting $y = \frac{v}{2}$, $\dot{y} = \frac{\dot{v}}{2}$, and $\phi = |y|$:

$$\dot{\phi} = \frac{y \cdot \dot{y}}{\phi} \tag{54}$$

$$\dot{q}_w = -\dot{\phi}\sin(\phi) = (y \cdot \dot{y})\frac{\sin(\phi)}{\phi} \tag{55}$$

$$\begin{aligned} \dot{q}_v = \frac{\sin(\phi)}{\phi}\dot{y} - \frac{\dot{\phi}\sin(\phi)}{\phi^2}y + \frac{\dot{\phi}\cos(\phi)}{\phi}y = \\ \frac{\sin(\phi)}{\phi}\dot{y} + \left(\frac{\cos(\phi) - \frac{\sin(\phi)}{\phi}}{\phi^2}\right)(\dot{y} \cdot y)\, y \end{aligned} \tag{56}$$

When ϕ goes to zero, we can approximate $\frac{\sin(\phi)}{\phi}$ with the series in (23) and the other singular factor as:

$$\frac{\cos(\phi) - \frac{\sin(\phi)}{\phi}}{\phi^2} = -\frac{1}{3} + \frac{\phi^2}{30} - \frac{\phi^4}{840} + \frac{\phi^6}{45360} + \ldots \tag{57}$$

E. Spherical Linear Interpolation

Spherical Linear Interpolation, *SLERP*, interpolates between two quaternions. SLERP can be understood geometrically by considering a relative orientation in the axis-angle form. Consider the relative quaternion q_r between two endpoints, $q_1 \otimes q_r = q_2$, given in axis angle form $(\hat{u}_r, \theta_r)$. To interpolate between q_1 and q_2, we apply the $q(\tau) = q_1 \otimes q_s(\tau)$, where q_s is a rotation about $\hat{u}_r$ with angle θ_s varying from 0 to θ_r as τ varies from 0 to 1. We can compute the rotation vector form of q_s from that of q_r as $v_s = \tau v_r$.

Composing definitions for quaternion and rotation vector conversion and quaternion exponents:

$$q(\tau) = q_1 \otimes \exp\left(\tau \ln\left({q_1}^* \otimes q_2\right)\right) = q_1 \otimes \left({q_1}^* \otimes q_2\right)^t \tag{58}$$

To interpolate in the shorter direction, e.g., $-\frac{\pi}{2}$ vs. $+\frac{3\pi}{2}$, scale ${q_1}^* \otimes q_2$ so it has a positive scalar element.

A more efficient computation for SLERP [2] is:

$$\phi = |\angle(\vec{q_1}, \vec{q_2})| \tag{59}$$

$$\theta = \begin{cases} \phi > \frac{\pi}{2} & \pi - \phi \\ \phi \leq \frac{\pi}{2} & \phi \end{cases} \tag{60}$$

$$q(\tau) = \begin{cases} \phi > \frac{\pi}{2} & \frac{\sin(\theta - \tau\theta)}{\sin(\theta)}q_1 - \frac{\sin(\tau\theta)}{\sin(\theta)}q_2 \\ \phi \leq \frac{\pi}{2} & \frac{\sin(\theta - \tau\theta)}{\sin(\theta)}q_1 + \frac{\sin(\tau\theta)}{\sin(\theta)}q_2 \end{cases} \tag{61}$$

F. Integration

Euler or Runge-Kutta integration of quaternion derivatives would not preserve the unit constraint, introducing error. We can instead integrate a constant rotational velocity with:

$$q_1 = \exp\left(\frac{\omega \Delta t}{2}\right) \otimes q_0 \tag{62}$$

$$= \exp\left(\Delta t \dot{q} \otimes q_0^*\right) \otimes q_0 \tag{63}$$

G. Finite Difference

Based on (62), we can compute a finite difference velocity ω_Δ between two orientations:

$$\omega_\Delta = 2\ln\left(q_1 \otimes q_0^*\right) \tag{64}$$

$$\dot{q}_\Delta = \ln\left(q_1 \otimes q_0^*\right) \otimes q_0 \tag{65}$$

IV. Dual Quaternions and Euclidean Transforms

Dual quaternions are convenient for representing Euclidean transformations. Formally, dual quaternions are the generalization of quaternions to dual numbers.

A. Dual Numbers

Dual numbers are similar to complex numbers, but the square of the dual element ε is zero:

$$\tilde{z} = a + b\varepsilon \tag{66}$$
$$\varepsilon \neq 0 \tag{67}$$
$$\varepsilon^2 = 0 \tag{68}$$

If we consider the Taylor series of $f(a+b\varepsilon)$ at point a, we obtain the following property:

$$f(a+b\varepsilon) = f(a) + bf'(a)\varepsilon \tag{69}$$

This lets us define a few functions for dual numbers:

$$\cos(a+b\varepsilon) = \cos(a) - \sin(a)\, b\varepsilon \tag{70}$$
$$\sin(a+b\varepsilon) = \sin(a) + \cos(a)\, b\varepsilon \tag{71}$$
$$\exp(a+b\varepsilon) = e^a + e^a b\varepsilon \tag{72}$$
$$\sqrt{a+b\varepsilon} = \sqrt{a} + \frac{b}{2\sqrt{a}}\varepsilon \tag{73}$$

B. Representation

Dual quaternions are quaternions with dual numbers for elements.

$$\begin{aligned} \mathcal{S} &= \\ \tilde{x}\mathit{i} + \tilde{y}\mathit{j} + \tilde{z}\mathit{k} + \tilde{w} &= \\ (r_x + d_x\varepsilon)\mathit{i} + (r_y + d_y\varepsilon)\mathit{j} + (r_z + d_z\varepsilon)\mathit{k} + (r_w + d_w\varepsilon) &= \\ (r_x\mathit{i} + r_y\mathit{j} + r_z\mathit{k} + r_w) + (d_x\mathit{i} + d_y\mathit{j} + d_z\mathit{k} + d_w)\varepsilon &= \\ \mathit{r} + \mathit{d}\varepsilon \end{aligned} \tag{74}$$

For computation, it is convenient to represent dual quaternion $\mathcal{S}$ factored into the separate real and dual parts r and d:

$$\begin{aligned} \mathcal{S} &= \mathit{r} + \mathit{d}\varepsilon \\ &= \mathcal{S}\left(\mathit{r},\ \mathit{d}\right) \end{aligned} \tag{75}$$

C. Construction

We can produce a dual quaternion for some transformation represented by the rotational quaternion q, and the translation vector v as follows:

$$\mathit{r} = \mathit{q} \tag{76}$$
$$\mathit{d} = \frac{1}{2}\mathit{v} \otimes \mathit{r} \tag{77}$$

Translation v is augmented with 0 as the scalar element for the quaternion multiply. The real part r represents orientation, and the dual part d represents translation. Note that the real part r will be a unit quaternion while the dual part d has no such restriction.

To extract the translation, we do:

$$v = 2\mathit{d} \otimes \mathit{r}^* \tag{78}$$

D. Multiplication

Multiplication is defined in terms of the standard quaternion multiply, performed over both real and dual parts:

$$\mathcal{A} \otimes \mathcal{B} = \mathcal{S}\left(\mathit{a}_r \otimes \mathit{b}_r,\quad \mathit{a}_r \otimes \mathit{b}_d + \mathit{a}_d \otimes \mathit{b}_r\right) \tag{79}$$

E. Matrix Form

We can also represent the dual quaternion multiplication as a matrix multiply. Based on (11):

$$\begin{aligned} \mathcal{A} \otimes \mathcal{B} = \begin{pmatrix} \mathit{a}_r \otimes \mathit{b}_r \\ \mathit{a}_r \otimes \mathit{b}_d + \mathit{a}_d \otimes \mathit{b}_r \end{pmatrix} &= \\ \mathbf{A_L}\vec{B} = \begin{bmatrix} \mathbf{A_{r,L}} & 0 \\ \mathbf{A_{d,L}} & \mathbf{A_{r,L}} \end{bmatrix} \vec{B} &= \\ \mathbf{B_R}\vec{A} = \begin{bmatrix} \mathbf{B_{r,R}} & 0 \\ \mathbf{B_{d,R}} & \mathbf{B_{r,R}} \end{bmatrix} \vec{A} \end{aligned} \tag{80}$$

F. Conjugate

$$\mathcal{S}^* = \mathcal{S}\left(\mathit{s}_r^*,\ \mathit{s}_d^*\right) \tag{81}$$

G. Exponential

We derive the dual quaternion exponential by expanding (22) using dual arithmetic:

$$\phi = |r_v| \tag{82}$$
$$k = r_v \cdot d_v \tag{83}$$
$$e^{\mathcal{S}} = e^{\tilde{w}}\,\mathcal{S}\left(\mathcal{H}\left(\frac{s}{\phi}r_v,\ c\right),\ \mathcal{H}\left(\frac{s}{\phi}d_v + \frac{c - \frac{s}{\phi}}{\phi^2}kr_v,\ -\frac{s}{\phi}k\right)\right) \tag{84}$$

where $\tilde{w} = r_w + d_w\varepsilon$.

Then, to handle the singularity at $\phi = 0$, we use (23) and:

$$\frac{\cos(\phi) - \frac{\sin(\phi)}{\phi}}{\phi^2} = -\frac{1}{3} + \frac{\phi^2}{30} - \frac{\phi^4}{840} + \frac{\phi^6}{45360} + \dots \tag{85}$$

H. Logarithm

We derive the dual quaternion logarithm by expanding (26) using dual arithmetic:

$$\phi = \operatorname{atan2}(|r_v|, r_w) \tag{86}$$
$$k = r_v \cdot d_v \tag{87}$$
$$\alpha = \frac{r_w - \frac{\phi}{|r_v|}|r|^2}{|r_v|^2} \tag{88}$$
$$(\ln S)_r = \mathcal{H}\left(\frac{\phi}{|r_v|}r_v,\ \ln|r|\right) \tag{89}$$
$$(\ln S)_d = \mathcal{H}\left(\frac{k\alpha - d_w}{|r|^2}r_v + \frac{\phi}{|r_v|}d_v,\ \frac{k + r_w d_w}{|r|^2}\right) \tag{90}$$

To handle the singularity at $|r_v| = 0$, we apply (27) and (28) to handle $\frac{\phi}{|r_v|}$. Then, we rewrite α as:

$$
\begin{aligned}
\frac{r_w - \frac{\phi}{|r_v|}|r|^2}{|r_v|^2} &= \\
\frac{r_w}{|r_v|^2} - \frac{\phi |r|^2}{|r_v|^3} &= \\
\frac{r_w |r|^2}{|r_v|^2 |r|^2} - \frac{\phi |r|^3}{|r_v|^3 |r|} &= \\
\frac{1}{|r|}\left(\frac{r_w}{|r|}\frac{|r|^2}{|r_v|^2} - \phi\frac{|r|^3}{|r_v|^3}\right) &= \\
\frac{1}{|r|}\left(\frac{\cos(\phi)}{\sin^2(\phi)} - \frac{\phi}{\sin^3(\phi)}\right) & \qquad (91)
\end{aligned}
$$

This gives the Taylor series:

$$\frac{c}{s^2} - \frac{\phi}{s^3} = -\frac{2}{3} - \frac{1}{5}\phi^2 - \frac{17}{420}\phi^4 - \frac{29}{4200}\phi^6 + \dots \qquad (92)$$

I. Chaining Transforms

Transforms are chained by multiplying the dual quaternions.

J. Transforming a point

We can transform a point v by constructing a dual quaternion for translation v and identity rotation, and chaining it onto the transform, then extracting the resulting translation:

$$\mathcal{S}' = \mathcal{S} \otimes \mathbb{S}\left(\mathcal{H}(0,\ 1),\ \frac{1}{2}v\right) \qquad (93)$$

$$v' = 2s'_d \otimes s'^*_r \qquad (94)$$

This reduces to:

$$v' = (2s_d + s_r \otimes v) \otimes s^*_r \qquad (95)$$

K. Derivatives

1) Product Rule: Because dual quaternion multiplication is a linear operation (see section B), the product rule applies:

$$\frac{d}{dt}(\mathcal{S}_1 \otimes \mathcal{S}_2) = \dot{\mathcal{S}}_1 \otimes \mathcal{S}_2 + \mathcal{S}_1 \otimes \dot{\mathcal{S}}_2 \qquad (96)$$

2) Angular Velocity: Angular velocity computation is identical to the single unit quaternion case:

$$\dot{r} = \frac{1}{2}\omega \otimes r \qquad (97)$$

$$\omega = 2\dot{r} \otimes r^* \qquad (98)$$

3) Translational Velocity: We find the equation for the derivative of the dual part by differentiating (77),

$$\dot{d} = \frac{1}{2}(\dot{v} \otimes r + v \otimes \dot{r}) \qquad (99)$$

Translational velocity comes from differentiating (78):

$$\dot{v} = 2(\dot{d} \otimes r^* + d \otimes (\dot{r})^*) \qquad (100)$$

Representation	Storage
Dual Quaternion	8
Implicit Dual Quaternion	7
Transformation Matrix	12

TABLE IV
STORAGE REQUIREMENTS FOR TRANSFORMATION REPRESENTATIONS

Representation	Chain	Transform
Dual Quaternion	48 multiply, 40 add	28 multiply 28 add
Implicit Dual Quaternion	31 multiply, 30	15 multiply, 18 add
Transformation Matrix	36 multiply, 27 add	9 multiply, 9 add

TABLE V
COMPUTATIONAL REQUIREMENTS FOR ORIENTATION REPRESENTATIONS

L. Integration

To integrate dual quaternions, we first introduce the *twist*, Ω:

$$\Omega = \mathbb{S}\left(\mathcal{H}(\omega,\ 0),\ \mathcal{H}(\dot{v} + v \times \omega,\ 0)\right) \qquad (101)$$

where ω is angular velocity, v is translation, and $\dot{v}$ is translational velocity.

Then, integration of a constant velocity is given by:

$$\mathcal{S}_1 = \exp\left(\frac{\Omega \Delta t}{2}\right) \otimes \mathcal{S}_0 \qquad (102)$$

V. IMPLICIT DUAL QUATERNIONS

We can implicitly represent the dual quaternion for a Euclidean transform by storing orientation quaternion r and translation vector v:

$$E = \mathbb{S}_{\mathrm{i}}\left(r,\ v\right) \qquad (103)$$

This form allows more efficient computation for some operations.

A. Chaining transforms

From dual quaternion multiplication (79), we derive the multiplication formula for the implicit form:

$$
\begin{aligned}
C_v = 2C_d \otimes C^*_r &= \\
2\left(A_r \otimes B_d + A_d \otimes B_r\right) \otimes \left(A_r \otimes B_r\right)^* &= \\
2\left(A_r \otimes \frac{B_v \otimes B_r}{2} + \frac{A_v \otimes A_r}{2} \otimes B_r\right) \otimes B^*_r \otimes A^*_r &= \\
\left(A_r \otimes B_v + A_v \otimes A_r\right) \otimes A^*_r &= \\
A_r \otimes B_v \otimes A^*_r + A_v &
\end{aligned}
$$

This is equivalent to rotating B_v by A_r, then adding A_v. Thus, we chain transforms with:

$$C_r = A_r \otimes B_r \qquad (104)$$

$$C_v = \operatorname{rot}(A_r, B_v) + A_v \qquad (105)$$

B. Transforming points

To transform point p, we first rotate it by the given orientation r, then add the translation v

$$p' = \operatorname{rot}(r, p) + v \qquad (106)$$

C. Conjugate

From the dual quaternion conjugate (81) for $S = (r, d)$:

$$\begin{aligned}(S^*)_v = 2(S^*)_d \otimes ((S^*)_r)^* &= \\ 2d^* \otimes (r^*)^* &= \\ 2(\frac{1}{2} v \otimes r)^* \otimes r &= \\ (v \otimes r)^* \otimes r &= \\ r^* \otimes v^* \otimes r &= \\ - \operatorname{rot}(r^*, v)\end{aligned}$$

Thus, to find the conjugate translation, we rotate v by r^* and negate.

D. Derivatives

The transform chaining in (105) is not linear, so we cannot apply the product rule. Instead, we directly differentiate (105):

$$\frac{d}{dt}\left(\mathcal{S}_\mathrm{i}\begin{pmatrix} r_1 \\ v_1 \end{pmatrix} \otimes \mathcal{S}_\mathrm{i}\begin{pmatrix} r_2 \\ v_2 \end{pmatrix}\right) = \mathcal{S}_\mathrm{i}\begin{pmatrix} \dot{r}_1 \otimes r_2 + r_1 \otimes \dot{r}_2 \\ \dot{v}_1 + \dot{q}_1 \otimes v_2 \otimes q_1^* + q_1 \otimes \dot{v}_2 \otimes q_1^* + q_1 \otimes v_2 \otimes \dot{q}_1^* \end{pmatrix} \quad (107)$$

VI. Matrices and Euclidean Transforms

A. Rotation Matrix

Using the matrix expansions of quaternion multipication, we can rewrite the quaternion rotation operator as a single matrix multiply:

$$q \otimes v \otimes q^* = \mathbf{Q_L} \vec{v} \otimes q^* = (\mathbf{Q}^*)_\mathbf{R} \mathbf{Q_L} \vec{v} = \mathbf{R}\vec{v} = \begin{bmatrix} -q_z^2 - q_y^2 + q_x^2 + q_w^2 & 2q_x q_y - 2q_z q_w & 2q_x q_z + 2q_y q_w \\ 2q_z q_w + 2q_x q_y & -q_z^2 + q_y^2 - q_x^2 + q_w^2 & 2q_y q_z - 2q_x q_w \\ 2q_x q_z - 2q_y q_w & 2q_y q_z + 2q_x q_w & q_z^2 - q_y^2 - q_x^2 + q_w^2 \end{bmatrix} v \quad (108)$$

The matrix R has geometric significance as well. The ith column of a R is the ith axis of the child frame in the parent frames coordinates.

B. Transformation Matrix

$$\mathbf{T} = \begin{bmatrix} \mathbf{R} & \mathbf{v} \\ 0 & 1 \end{bmatrix} \quad (109)$$

C. Transforming Points

$$\begin{bmatrix} p' \\ 1 \end{bmatrix} = \mathbf{T}\vec{p} = \begin{bmatrix} \mathbf{T_R}\vec{p} + \mathbf{T_v} \\ 1 \end{bmatrix} \quad (110)$$

D. Chaining Transforms

$$\mathbf{C} = \mathbf{AB} = \begin{bmatrix} \mathbf{R_A R_B} & (\mathbf{R_A v_B} + \mathbf{v_A}) \\ 0 & 1 \end{bmatrix} \quad (111)$$

References

[1] KAHAN, W. How futile are mindless assessments of roundoff in floating-point computation. Tech. rep., U.C. Berkeley, 2006. http://www.cs.berkeley.edu/~wkahan/Mindless.pdf.

[2] SHOEMAKE, K. Animating rotation with quaternion curves. *ACM SIGGRAPH computer graphics 19*, 3 (1985), 245–254.

Glossary

axis-angle
Rotation representation $(\hat{u}, \theta)$, where $\hat{u}$ is a unit vector representing an axis of rotation and θ is an angle to rotate about $\hat{u}$. 5

dual number
Number with dual element ε, where $\varepsilon^2 = 0$. 5

pure quaternion
A quaternion with zero scalar part. 3, 4

rotation vector
Scaled form of the axis-angle representation, $v = \theta\hat{u}$. 5

scalar
The real part of the quaternion, i.e., the w element. 2, 3

SIMD
Single Instruction Multiple Data. Type of CPU instructions which perform multiple computations with a single instruction, such as element-wise addition or multiplication of several values. 2, 5

unit quaternion
A quaternion with norm of one. 3, 4

vector
The imaginary part of the quaternion, i.e., the x, y, and z elements. 2, 3

Appendix A
History

Quaternions were invented in the mid-nineteenth century by William Rowan Hamilton, who spent the rest of his life exploring their properties. They quickly found use among physicists; Maxwell's equations were originally formulated using quaternions.

Around the turn of the twentieth century, Josiah Gibbs published his *Vector Analysis*, presented as a simplification over quaternions. The chief distinction was the invention of the dot and cross product operators, splitting quaternion multiplication into two separate operations. Eventually, Gibbs's notation overtook quaternions as the representation of choice among physicists and engineers.

Though quaternions may have lost the overall popularity contest to Gibbs's vector analysis, their useful numerical properties mean quaternions still have some role to play.

Appendix B
Derivation of Quaternion Multiplication

First, the basis elements axiom:

$$\imath^2 = \jmath^2 = \kappa^2 = \imath\jmath\kappa = -1$$

A. Derivation of Quaternion Basis Equalities

$$
\begin{aligned}
ijk &= -1 \\
ijkk &= -k \\
-ij &= -k \\
ij &= k \\
iij &= ik \\
-j &= ik \\
-jj &= jik \\
1 &= jik \\
k &= jikk \\
k &= -ji \\
ki &= -jii \\
ki &= j \\
jki &= jj \\
jki &= -1 \\
jkii &= -i \\
jk &= i \\
jkk &= ik \\
-j &= ik \\
-jj &= ikj \\
1 &= ikj \\
i &= iikj \\
i &= -kj
\end{aligned}
$$

B. Derivation of Quaternion Multiplication

1) Multiply the two quaternions:

$$p \otimes q = (p_w + p_x i + p_y j + p_z k)(q_w + q_x i + q_y j + q_z k)$$

2) Distribute terms of q_1 over terms of q_2:

$$
\begin{aligned}
\implies & p_w(q_w + q_x i + q_y j + q_z k)+ \\
& p_x i(q_w + q_x i + q_y j + q_z k)+ \\
& p_y j(q_w + q_x i + q_y j + q_z k)+ \\
& p_z k(q_w + q_x i + q_y j + q_z k)
\end{aligned}
$$

3) Distribute again:

$$
\begin{aligned}
\implies & p_w q_w + p_w q_x i + p_w q_y j + p_w q_z k+ \\
& p_x q_w i + p_x q_x i^2 + p_x q_y ij + p_x q_z ik+ \\
& p_y q_w j + p_y q_x ji + p_y q_y j^2 + p_y q_z jk+ \\
& p_z q_w k + p_z q_x ki + p_z q_y kj + p_z q_z k^2
\end{aligned}
$$

4) Simplify basis elements again:

$$
\begin{aligned}
\implies & p_w q_w + p_w q_x i + p_w q_y j + p_w q_z k+ \\
& p_x q_w i - p_x q_x + p_x q_y k - p_x q_z j+ \\
& p_y q_w j - p_y q_x k - p_y q_y + p_y q_z i+ \\
& p_z q_w k + p_z q_x j - p_z q_y i - p_z q_z
\end{aligned}
$$

5) Combine terms by basis element:

$$
\begin{aligned}
\implies & (p_w q_x + p_x q_w + p_y q_z - p_z q_y)i+ \\
& (p_w q_y - p_x q_z + p_y q_w + p_z q_x)j+ \\
& (p_w q_z + p_x q_y - p_y q_x + p_z q_w)k+ \\
& (p_w q_w - p_x q_x - p_y q_y - p_z q_z)
\end{aligned}
$$

6) Reorder the terms:

$$
\begin{aligned}
\implies & (p_y q_z - p_z q_y + p_w q_x + q_w p_x)i+ \\
& (p_z q_x - p_x q_z + p_w q_y + q_w p_y)j+ \\
& (p_x q_y - p_y q_x + p_w q_z + q_w p_z)k+ \\
& (p_w q_w - p_x q_x - p_y q_y - p_z q_z) \\
= & \begin{pmatrix} p_v \times q_v + p_w q_v + q_v p_v \\ p_w q_w - p_v \cdot q_v \end{pmatrix}
\end{aligned}
$$