Quaternion Algebra and Calculus

David Eberly
Geometric Tools, LLC
http://www.geometrictools.com/
Copyright © 1998-2016. All Rights Reserved.

Created: March 2, 1999
Last Modified: August 18, 2010

Contents

1 Quaternion Algebra 2
2 Relationship of Quaternions to Rotations 3
3 Quaternion Calculus 5
4 Spherical Linear Interpolation 6
5 Spherical Cubic Interpolation 7
6 Spline Interpolation of Quaternions 8
This document provides a mathematical summary of quaternion algebra and calculus and how they relate to rotations and interpolation of rotations. The ideas are based on the article [1].

1 Quaternion Algebra

A quaternion is given by \( q = w + xi + yj + zk \) where \( w, x, y, \) and \( z \) are real numbers. Define \( q_n = w_n + x_n i + y_n j + z_n k \) (\( n = 0, 1 \)). Addition and subtraction of quaternions is defined by

\[
q_0 \pm q_1 = (w_0 \pm x_0 i + y_0 j + z_0 k) \pm (w_1 + x_1 i + y_1 j + z_1 k) = (w_0 \pm w_1) + (x_0 \pm x_1)i + (y_0 \pm y_1)j + (z_0 \pm z_1)k. \tag{1}
\]

Multiplication for the primitive elements \( i, j, \) and \( k \) is defined by \( i^2 = j^2 = k^2 = -1, \) \( ij = -ji = k, \) \( jk = -kj = i, \) and \( ki = -ik = j. \) Multiplication of quaternions is defined by

\[
q_0 q_1 = (w_0 + x_0 i + y_0 j + z_0 k)(w_1 + x_1 i + y_1 j + z_1 k) = (w_0 w_1 - x_0 x_1 - y_0 y_1 - z_0 z_1) + (w_0 x_1 + x_0 w_1 + y_0 z_1 - z_0 y_1)i + (w_0 y_1 - x_0 z_1 + y_0 w_1 + z_0 x_1)j + (w_0 z_1 + x_0 y_1 - y_0 x_1 + z_0 w_1)k. \tag{2}
\]

Multiplication is not commutative in that the products \( q_0 q_1 \) and \( q_1 q_0 \) are not necessarily equal.

The conjugate of a quaternion is defined by

\[
q^* = (w + xi + yj + zk)^* = w - xi - yj - zk. \tag{3}
\]

The conjugate of a product of quaternions satisfies the properties \((p^*)^* = p\) and \((pq)^* = q^*p^*\).

The norm of a quaternion is defined by

\[
N(q) = N(w + xi + yj + zk) = w^2 + x^2 + y^2 + z^2. \tag{4}
\]

The norm is a real-valued function and the norm of a product of quaternions satisfies the properties \(N(q^*) = N(q)\) and \(N(pq) = N(p)N(q)\).

The multiplicative inverse of a quaternion \( q \) is denoted \( q^{-1} \) and has the property \( qq^{-1} = q^{-1}q = 1 \). It is constructed as

\[
q^{-1} = q^* / N(q) \tag{5}
\]

where the division of a quaternion by a real-valued scalar is just componentwise division. The inverse operation satisfies the properties \((p^{-1})^{-1} = p\) and \((pq)^{-1} = q^{-1}p^{-1}\).

A simple but useful function is the selection function

\[
W(q) = W(w + xi + yj + zk) = w \tag{6}
\]

which selects the “real part” of the quaternion. This function satisfies the property \( W(q) = (q + q^*)/2 \).
The quaternion \( q = w + xi + yj + zk \) may also be viewed as \( q = w + \hat{v} \) where \( \hat{v} = xi + yj + zk \). If we identify \( \hat{v} \) with the 3D vector \((x, y, z)\), then quaternion multiplication can be written using vector dot product (\( \cdot \)) and cross product (\( \times \)) as

\[
(w_0 + \hat{v}_0)(w_1 + \hat{v}_1) = (w_0w_1 - \hat{v}_0 \cdot \hat{v}_1) + w_0\hat{v}_1 + w_1\hat{v}_0 + \hat{v}_0 \times \hat{v}_1.
\]

In this form it is clear that \( q_0q_1 = q_1q_0 \) if and only if \( \hat{v}_0 \times \hat{v}_1 = 0 \) (these two vectors are parallel).

A quaternion \( q \) may also be viewed as a 4D vector \((w, x, y, z)\). The dot product of two quaternions is

\[
q_0 \cdot q_1 = w_0w_1 + x_0x_1 + y_0y_1 + z_0z_1 = W(q_0q_1^*).\]

(8)

A unit quaternion is a quaternion \( q \) for which \( N(q) = 1 \). The inverse of a unit quaternion and the product of unit quaternions are themselves unit quaternions. A unit quaternion can be represented by

\[
q = \cos \theta + \hat{u} \sin \theta
\]

(9)

where \( \hat{u} \) as a 3D vector has length 1. However, observe that the quaternion product \( \hat{u}\hat{u} = -1 \). Note the similarity to unit length complex numbers \( \cos \theta + i \sin \theta \). In fact, Euler’s identity for complex numbers generalizes to quaternions,

\[
\exp(\hat{u}\theta) = \cos \theta + \hat{u} \sin \theta,
\]

(10)

where the exponential on the left-hand side is evaluated by symbolically substituting \( \hat{u}\theta \) into the power series representation for \( \exp(x) \) and replacing products \( \hat{u}\hat{u} \) by \(-1\). From this identity it is possible to define the power of a unit quaternion,

\[
q^t = (\cos \theta + \hat{u} \sin \theta)^t = \exp(t\hat{u}\theta) = \cos(t\theta) + \hat{u} \sin(t\theta).
\]

(11)

It is also possible to define the logarithm of a unit quaternion,

\[
\log(q) = \log(\cos \theta + \hat{u} \sin \theta) = \log(\exp(\hat{u}\theta)) = \hat{u}\theta.
\]

(12)

It is important to note that the noncommutativity of quaternion multiplication disallows the standard identities for exponential and logarithm functions. The quaternions \( \exp(p) \exp(q) \) and \( \exp(p + q) \) are not necessarily equal. The quaternions \( \log(pq) \) and \( \log(p) + \log(q) \) are not necessarily equal.

2 Relationship of Quaternions to Rotations

A unit quaternion \( q = \cos \theta + \hat{u} \sin \theta \) represents the rotation of the 3D vector \( \hat{v} \) by an angle \( 2\theta \) about the 3D axis \( \hat{u} \). The rotated vector, represented as a quaternion, is \( R(\hat{v}) = q\hat{v}q^* \). The proof requires showing that \( R(\hat{v}) \) is a 3D vector, a length-preserving function of 3D vectors, a linear transformation, and does not have a reflection component.
To see that \( R(\hat{v}) \) is a 3D vector,
\[
W(R(\hat{v})) = W(q\hat{v}q^*)
= [(q\hat{v}q^*) + (q\hat{v}q^*)]/2
= [q\hat{v}q^* + q\hat{v}q^*]/2
= q((\hat{v} + \hat{v}^*)/2)q^*
= qW(\hat{v})q^*
= W(\hat{v})
= 0.
\]

To see that \( R(\hat{v}) \) is length-preserving,
\[
N(R(\hat{v})) = N(q\hat{v}q^*)
= N(q)N(\hat{v})N(q^*)
= N(q)N(\hat{v})N(q)
= N(\hat{v}).
\]

To see that \( R(\hat{v}) \) is a linear transformation, let \( a \) be a real-valued scalar and let \( \hat{v} \) and \( \hat{w} \) be 3D vectors; then
\[
R(a\hat{v} + \hat{w}) = q(a\hat{v} + \hat{w})q^*
= (qa\hat{v}q^*) + (q\hat{w}q^*)
= a(q\hat{v}q^*) + (q\hat{w}q^*)
= aR(\hat{v}) + R(\hat{w}),
\]
thereby showing that the transform of a linear combination of vectors is the linear combination of the transforms.

The previous three properties show that \( R(\hat{v}) \) is an orthonormal transformation. Such transformations include rotations and reflections. Consider \( R \) as a function of \( q \) for a fixed vector \( \hat{v} \). That is, \( R(q) = q\hat{v}q^* \). This function is a continuous function of \( q \). For each \( q \) it is a linear transformation with determinant \( D(q) \), so the determinant itself is a continuous function of \( q \). Thus, \( \lim_{q \to 1} R(q) = R(1) = I \), the identity function (the limit is taken along any path of quaternions which approach the quaternion 1) and \( \lim_{q \to 1} D(q) = D(1) = 1 \).

By continuity, \( D(q) \) is identically 1 and \( R(q) \) does not have a reflection component.

Now we prove that the unit rotation axis is the 3D vector \( \hat{u} \) and the rotation angle is \( 2\theta \). To see that \( \hat{u} \) is a unit rotation axis we need only show that \( \hat{u} \) is unchanged by the rotation. Recall that \( \hat{u}^2 = \hat{u}\hat{u} = -1 \). This implies that \( \hat{u}^3 = -\hat{u} \). Now
\[
R(\hat{u}) = q\hat{u}q^*
= (\cos \theta + \hat{u}\sin \theta)\hat{u}(\cos \theta - \hat{u}\sin \theta)
= (\cos \theta)^2\hat{u} - (\sin \theta)^2\hat{u}^3
= (\cos \theta)^2\hat{u} - (\sin \theta)^2(\hat{u})
= \hat{u}.
\]
To see that the rotation angle is $2\theta$, let $\hat{u}, \hat{v},$ and $\hat{w}$ be a right-handed set of orthonormal vectors. That is, the vectors are all unit length; $\hat{u} \cdot \hat{v} = \hat{u} \cdot \hat{w} = \hat{v} \cdot \hat{w} = 0$; and $\hat{u} \times \hat{v} = \hat{w}, \hat{v} \times \hat{w} = \hat{u},$ and $\hat{w} \times \hat{u} = \hat{v}$. The vector $\hat{v}$ is rotated by an angle $\phi$ to the vector $q \hat{v} q^*$, so $\hat{v} \cdot (q \hat{v} q^*) = \cos(\phi)$. Using equation (8) and $\hat{v}^* = -\hat{v}$, and $\hat{p}^2 = -1$ for unit quaternions with zero real part,

$$
\cos(\phi) = \hat{v} \cdot (q \hat{v} q^*) = W(\hat{v}^* q \hat{v} q^*)
= W[-\hat{v}(\cos \theta + \hat{u} \sin \theta)\hat{v}(\cos \theta - \hat{u} \sin \theta)]
= W[(-\hat{v} \cos \theta - \hat{u} \sin \theta)(\hat{v} \cos \theta - \hat{u} \sin \theta)]
= W[-\hat{v}^2(\cos \theta)^2 + \hat{v}^2 \hat{u} \sin \theta \cos \theta - \hat{u} \hat{v} \hat{v} \sin \theta \cos \theta + (\hat{v} \hat{u})^2(\sin \theta)^2]
= W[(\cos \theta)^2 - (\sin \theta)^2 - (\hat{u} + \hat{v} \hat{u}) \sin \theta \cos \theta]
$$

Now $\hat{v} \hat{u} = -\hat{v} \cdot \hat{u} + \hat{v} \times \hat{u} = \hat{v} \times \hat{u} = -\hat{u} \hat{v} = \hat{w} \cdot \hat{v} - \hat{w} \times \hat{v} = \hat{u}$. Consequently,

$$
\cos(\phi) = W[(\cos \theta)^2 - (\sin \theta)^2 - (\hat{u} + \hat{v} \hat{u}) \sin \theta \cos \theta]
= W[(\cos \theta)^2 - (\sin \theta)^2 - 2 \sin \theta \cos \theta]
= (\cos \theta)^2 - (\sin \theta)^2
= \cos(2\theta)
$$

and the rotation angle is $\phi = 2\theta$.

It is important to note that the quaternions $q$ and $-q$ represent the same rotation since $(-q)\hat{v}(-q)^* = q \hat{v} q^*$. While either quaternion will do, the interpolation methods require choosing one over the other.

### 3 Quaternion Calculus

The only support we need for quaternion interpolation is to differentiate unit quaternion functions raised to a real-valued power. These formulas are identical to those derived in a standard calculus course, but the order of multiplication must be observed.

The derivative of the function $q^t$ where $q$ is a constant unit quaternion is

$$
\frac{d}{dt} q^t = q^t \log(q) \tag{13}
$$

where $\log$ is the function defined earlier by $\log(\cos \theta + \hat{u} \sin \theta) = \hat{u} \theta$. To prove this, observe that

$$
q^t = \cos(t\theta) + \hat{u} \sin(t\theta)
$$

in which case

$$
\frac{d}{dt} q^t = -\sin(t\theta)\theta + \hat{u} \cos(t\theta)\theta = \hat{u} \hat{u} \sin(t\theta)\theta + \hat{u} \cos(t\theta)\theta
$$

where we have used $-1 = \hat{u} \hat{u}$. Factoring this, we have

$$
\frac{d}{dt} q^t = (\hat{u} \sin(t\theta) + \cos(t\theta))\hat{u} \theta = q^t \log(q)
$$
The right-hand side also factors as log\(q\). Generally, the order of operands in a quaternion multiplication is important, but not in this special case. The power can be a function itself,\[
\frac{d}{dt} q^{f(t)} = f'(t)q^{f(t)} \log(q) \tag{14}\]
The method of proof is the same as that of the previous case where \(f(t) = t\).

Generally, a quaternion function may be written as
\[q(t) = \cos(\theta(t)) + \hat{u}(t) \sin(\theta(t))\tag{15}\]
where the angle \(\theta\) and \(\hat{u}\) both vary with \(t\). The derivative is
\[q'(t) = -\sin(\theta(t))\theta'(t) + \hat{u}(t) \cos(\theta(t))\theta'(t) + \hat{u}'(t) \sin(\theta(t)) = q(t)\hat{u}(t)\theta'(t) + \hat{u}'(t) \sin(\theta(t))\tag{16}\]
Because \(-1 = \hat{u}(t)\hat{u}(t)\), we also know that
\[0 = \hat{u}(t)\hat{u}'(t) + \hat{u}'(t)\hat{u}(t)\tag{17}\]
If you write \(\hat{u} = xi + yj + zk\) and expand the right-hand side of Equation (17), the equation becomes \(xx' + yy' + zz' = 0\). This implies the vectors \(u = (x, y, z)\) and \(u' = (x', y', z')\) are perpendicular. From this discussion, it is easily shown that
\[\hat{u}(t)q'(t) + q'(t)\hat{u}(t) = -2\theta'(t)q(t)\tag{18}\]

Now define \(h(t) = q(t)^{f^+(t)} = \cos(f(t)\theta(t)) + \hat{u}(t) \sin(f(t)\theta(t))\) \tag{19}\]
where \(q(t) = \cos(\theta(t)) + \hat{u}(t) \sin(\theta(t))\). The motivation for the definition is that we know how to compute \(q(t)^{f(s)}\) for independent variables \(s\) and \(t\), and we want this to be jointly continuous in the sense that \(q(t)^{f(t)} = \lim_{s \to t} q(t)^{f(s)}\). The derivative is
\[h'(t) = [-\sin(f\theta) + \hat{u}(t) \cos(f\theta)](f\theta)' + \hat{u}'(t) \sin(f\theta) = (\hat{u}h)(f\theta)' + \hat{u}'(f\theta)\tag{20}\]
Similar to Equation (18), it may be shown that
\[\hat{u}(t)h'(t) + h'(t)\hat{u}(t) = -2\frac{d[f(t)\theta(t)]}{dt} h(t)\]
Note that this last equation by itself is not enough information to completely determine \(h'(t)\), so consider it a sufficient condition for the derivative \(h'(t)\).

4 Spherical Linear Interpolation

The previous version of the document had a construction for spherical linear interpolation of two quaternions \(q_0\) and \(q_1\) treated as unit length vectors in 4-dimensional space, the angle \(\theta\) between them acute. The idea was that \(q(t) = c_0(t)q_0 + c_1(t)q_1\) where \(c_0(t)\) and \(c_1(t)\) are real-valued functions for \(0 \leq t \leq 1\) with \(c_0(0) = 1\), \(c_1(0) = 0\), \(c_0(1) = 0\), and \(c_1(1) = 1\). The quantity \(q(t)\) is required to be a unit vector, so \(1 = q(t) \cdot q(t) = c_0(t)^2 + 2 \cos(\theta)c_0(t)c_1(t) + c_1(t)^2\). This is the equation of an ellipse that I factored using methods of analytic geometry to obtain formulas for \(c_0(t)\) and \(c_1(t)\).
A simpler construction uses only trigonometry and solving two equations in two unknowns. As \( t \) uniformly varies between 0 and 1, the values \( q(t) \) are required to uniformly vary along the circular arc from \( q_0 \) to \( q_1 \). That is, the angle between \( q(t) \) and \( q_0 \) is \( \cos(t\theta) \) and the angle between \( q(t) \) and \( q_1 \) is \( \cos((1-t)\theta) \). Dotting the equation for \( q(t) \) with \( q_0 \) yields
\[
\cos(t\theta) = c_0(t) + \cos(\theta)c_1(t)
\]
and dotting the equation with \( q_1 \) yields
\[
\cos((1-t)\theta) = \cos(\theta)c_0(t) + c_1(t).
\]
These are two equations in the two unknowns \( c_0 \) and \( c_1 \). The solution for \( c_0 \) is
\[
c_0(t) = \frac{\cos(t\theta) - \cos(\theta)\cos((1-t)\theta)}{1 - \cos^2(\theta)} = \frac{\sin((1-t)\theta)}{\sin(\theta)}.
\]
The last equality is obtained by applying double-angle formulas for sine and cosine. By symmetry, \( c_1(t) = c_0(1-t) \). Or solve the equations for
\[
c_1(t) = \frac{\cos((1-t)\theta) - \cos(\theta)\cos(t\theta)}{1 - \cos^2(\theta)} = \frac{\sin(t\theta)}{\sin(\theta)}.
\]
The spherical linear interpolation, abbreviated as \( \text{slerp} \), is defined by
\[
\text{Slerp}(t; q_0, q_1) = \frac{q_0\sin((1-t)\theta) + q_1\sin(t\theta)}{\sin(\theta)}
\]
for \( 0 \leq t \leq 1 \).

Although \( q_1 \) and \( -q_1 \) represent the same rotation, the values of \( \text{Slerp}(t; q_0, q_1) \) and \( \text{Slerp}(t; q_0, -q_1) \) are not the same. It is customary to choose the sign \( \sigma \) on \( q_1 \) so that \( q_0 \bullet (\sigma q_1) \geq 0 \) (the angle between \( q_0 \) and \( \sigma q_1 \) is acute). This choice avoids extra spinning caused by the interpolated rotations.

For unit quaternions, \( \text{slerp} \) can be written as
\[
\text{Slerp}(t; q_0, q_1) = q_0(q_0^{-1}q_1)^t.
\]
The idea is that \( q_1 = q_0(q_0^{-1}q_1) \). The term \( q_0^{-1}q_1 = \cos \theta + \hat{u}\sin \theta \) where \( \theta \) is the angle between \( q_0 \) and \( q_1 \). The time parameter can be introduced into the angle so that the adjustment of \( q_0 \) varies uniformly with over the great arc between \( q_0 \) and \( q_1 \). That is, \( q(t) = q_0[\cos(t\theta) + \hat{u}\sin(t\theta)] = q_0[\cos \theta + \hat{u}\sin \theta]^t = q_0(q_0^{-1}q_1)^t \).

The derivative of \( \text{slerp} \) in the form of equation (22) is a simple application of equation (13),
\[
\text{Slerp}'(t; q_0, q_1) = q_0(q_0^{-1}q_1)^t \log(q_0^{-1}q_1).
\]

5 Spherical Cubic Interpolation

Cubic interpolation of quaternions can be achieved using a method described in [2] which has the flavor of bilinear interpolation on a quadrilateral. The evaluation uses an iteration of three \( \text{slerps} \) and is similar to the de Casteljau algorithm (see [3]). Imagine four quaternions \( p, a, b, \) and \( q \) as the ordered vertices of a quadrilateral. Interpolate \( c \) along the “edge” from \( p \) to \( q \) using \( \text{slerp} \). Interpolate \( d \) along the “edge” from
a to b. Now interpolate the edge interpolations c and d to get the final result e. The end result is denoted squad and is given by

\[
\text{Squad}(t; p, a, b, q) = \text{Slerp}(2t(1 - t); \text{Slerp}(t; p, q), \text{Slerp}(t; a, b))
\]

(24)

For unit quaternions we can use equation (22) to obtain a similar formula for squad,

\[
\text{Squad}(t; p, a, b, q) = \text{Slerp}(t; p, q)(\text{Slerp}(t; p, q)^{-1} \text{Slerp}(t; a, b))^{2t(1-t)}
\]

(25)

The derivative of squad in equation (25) is computed as follows. To simplify the notation, define \( U(t) = \text{Slerp}(t; p, q) \) and \( V(t) = \text{Slerp}(t; a, b) \). Equation (13) implies \( U'(t) = U(t) \log(p^{-1}q) \) and \( V'(t) = V(t) \log(a^{-1}b) \). Define \( W(t) \), \( \dot{\alpha}(t) \), and \( \phi(t) \) by

\[
W(t) = U(t)^{-1}V(t) = \cos(\phi(t)) + \dot{\alpha}(t)\sin(\phi(t))
\]

(26)

It is simple to see that \( U(t)W(t) = V(t) \). The derivative of \( W(t) \) is implicit in \( U(t)W'(t) + U'(t)W(t) = V'(t) \). The squad function is then

\[
\text{Squad}(t; p, a, b, q) = U(t)W(t)^{2t(1-t)}
\]

(27)

and its derivative is computed as shown next, using Equation (20),

\[
\text{Squad}'(t; p, q, a, b) = \frac{d}{dt} \left[ U W^{2t(1-t)} \right]
\]

\[
= U(t) \frac{d}{dt} \left[ W(t)^{2t(1-t)} \right] + U'(t) \left[ W(t)^{2t(1-t)} \right]
\]

\[
= U(t) \left\{ \frac{\dot{\alpha}(t)W(t)^{2t(1-t)}}{U(t)} [2t(1-t)\phi'(t) + (2 - 4t)\phi(t)] + \phi'(t) \sin(2t(1-t)\phi(t)) \right\}
\]

\[
+ U'(t)W(t)^{2t(1-t)}
\]

(28)

For spline interpolation using squad we will need to evaluate the derivative of squad at \( t = 0 \) and \( t = 1 \). Observe that \( U(0) = p, U'(0) = p \log(p^{-1}q), U(1) = q, U'(1) = q \log(p^{-1}q), V(0) = a, V'(0) = a \log(a^{-1}b), V'(1) = b, \) and \( V(1) = b \log(a^{-1}b) \). Also observe that \( \log(W(t)) = \dot{\alpha}(t)\phi(t) \) so that \( \log(p^{-1}a) = \log(W(0)) = \dot{\alpha}(0)\phi(0) \) and \( \log(q^{-1}b) = \log(W(1)) = \dot{\alpha}(1)\phi(1) \). The derivatives of squad at the endpoints are

\[
\text{Squad}'(0; p, a, b, q) = U(0) \{ \dot{\alpha}(0)[+2\phi(0)] \} + U'(0) = p[\log(p^{-1}q) + 2 \log(p^{-1}a)]
\]

\[
\text{Squad}'(1; p, a, b, q) = U(1) \{ \dot{\alpha}(1)[-2\phi(1)] \} + U'(1) = q[\log(p^{-1}q) - 2 \log(q^{-1}b)]
\]

(29)

6 Spline Interpolation of Quaternions

Given a sequence of \( N \) unit quaternions \( \{ q_n \}_{n=0}^{N-1} \), we want to build a spline which interpolates those quaternions subject to the conditions that the spline pass through the control points and that the derivatives are continuous. The idea is to choose intermediate quaternions \( a_n \) and \( b_n \) to allow control of the derivatives at the endpoints of the spline segments. More precisely, let \( S_n(t) = \text{Squad}(t; q_n, a_n, b_{n+1}, q_{n+1}) \) be the spline segments. By definition of squad it is easily shown that

\[
S_{n-1}(1) = q_n = S_n(0).
\]
To obtain continuous derivatives at the endpoints we need to match the derivatives of two consecutive spline segments,

\[ S'_{n-1}(1) = S'_n(0). \]

It can be shown from equation (29) that

\[ S'_{n-1}(1) = q_n \log(q_{n-1}^{-1}q_n) - 2 \log(q_n^{-1}b_n) \]

and

\[ S'_n(0) = q_n \log(q_n^{-1}q_{n+1}) + 2 \log(q_n^{-1}a_n). \]

The derivative continuity equation provides one equation in the two unknowns \( a_n \) and \( b_n \), so we have one degree of freedom. As suggested in [1], a good choice for the derivative at the control point uses an average \( T_n \) of “tangents”, so \( S'_{n-1}(1) = q_n T_n = S'_n(0) \) where

\[ T_n = \frac{\log(q_{n-1}^{-1}q_{n+1}) + \log(q_{n-1}^{-1}q_n)}{2}. \]  

(30)

We now have two equations to determine \( a_n \) and \( b_n \). Some algebra will show that

\[ a_n = b_n = q_n \exp\left(-\frac{\log(q_{n-1}^{-1}q_{n+1}) + \log(q_{n-1}^{-1}q_n)}{4}\right). \]  

(31)

Thus, \( S_n(t) = \text{Squad}(t; q_n, a_n, a_{n+1}, q_{n+1}) \).

EXAMPLE. To illustrate the cubic nature of the interpolation, consider a sequence of quaternions whose general term is \( q_n = \exp(i\theta_n) \). This is a sequence of complex numbers whose products do commute and for which the usual properties of exponents and logarithms do apply. The intermediate terms are

\[ a_n = \exp(-i(\theta_{n+1} - 6\theta_n + \theta_{n-1})/4). \]

Also,

\[ \text{Slerp}(t, q_n, q_{n+1}) = \exp(i((1-t)\theta_n + t\theta_{n+1})) \]

and

\[ \text{Slerp}(t, a_n, a_{n+1}) = \exp(-i((1-t)(\theta_{n+1} - 6\theta_n + \theta_{n-1}) + t(\theta_{n+2} - 6\theta_{n+1} + \theta_n))/4). \]

Finally,

\[ \text{Squad}(t, q_n, a_n, a_{n+1}, q_{n+1}) = \exp\left( [(1 - 2t)(1-t)][(1-t)\theta_n + t\theta_{n+1}] - [2(1-t)/4][(1-t)(\theta_{n+1} - 6\theta_n + \theta_{n-1}) + t(\theta_{n+2} - 6\theta_{n+1} + \theta_n)] \right). \]

The angular cubic interpolation is

\[ \phi(t) = -\frac{1}{2} t^2 (1-t)\theta_{n+2} + \frac{1}{2} t(2+2(1-t) - 3(1-t)^2)\theta_{n+1} + \frac{1}{2} (1-t)(2+2t-3t^2)\theta_n - \frac{1}{2} t(1-t)^2\theta_{n-1}. \]

It can be shown that \( \phi(0) = \theta_n \), \( \phi(1) = \theta_{n+1} \), \( \phi'(0) = (\theta_{n+1} - \theta_{n-1})/2 \), and \( \phi'(1) = (\theta_{n+2} - \theta_{n})/2 \). The derivatives at the end points are centralized differences, the average of left and right derivatives as expected.
References

