Quaternions Algebra, Their Applications in Rotations and Beyond Quaternions

Gökmen Günaşıtı

September 3, 2016

Abstract

The theory of quaternions was discovered in the middle of nineteenth century and they were commonly used to represent rotations. This thesis is written to review the basic properties of quaternions algebra and their applications in representing rotation of a body in 3-dimensional Euclidean space. Also, last sections in this thesis explore why the use of quaternions are more advantages than Euler angle sequences and can quaternions themselves be further generalized to another number systems?

1 Introduction

Remember complex numbers. These numbers are an extension of the real number system and the quaternions are four dimensional extension of this complex number system. Quaternions algebra are commonly used as a method for a body rotation in three-dimensional space. Although, Euler angle sequences are relatively easy to understand and traditional way to represent rotations, quaternions have an important advantage. When using Euler angle sequences, a phenomenon which is called Gimbal Lock (singularity) occurs when two axes align and a degree of freedom is lost. But, singularity is avoided when quaternions are used. Also, quaternions provide smooth rotation for a rigid body.

This thesis is arranged as follows. The next section describes historical context of quaternions, what are quaternions and how they emerged. Section 3 introduces an overview of quaternions algebra; how they define, what are the fundamental properties of quaternions and how they are written in alternative ways. Before giving a relationship of quaternions to rotations in 3D space, section 4 will give a brief overview of Euler angles and rotation
matrices and how they are used to rotate objects in $\mathbb{R}^3$. Also, this section describes gimbal lock phenomena using Euler angles. Section 5 includes a comparison between quaternions and Euler angles; what are the advantages and disadvantages of using quaternions than Euler angles. This section also introduces the spherical linear interpolation method. Before the last section, the thesis explores the generalization of quaternions and gives an overview about it in Section 6. The thesis is finally concluded in Section 7.

2 Historical Context

Quaternions were discovered by Sir William Rowan Hamilton (1805 - 1865), an Irish mathematician, in 1843. Hamilton wanted to generalize complex numbers in some way that would be applicable to three-dimensional (3D) space. Because, complex numbers have two parts, one part is an ordinary real number and one part is imaginary. At his first step, Hamilton wanted to extend the complex numbers to a new algebraic structure with each element consisting of one real part and two distinct imaginary parts. This would be known as the “theory of triplets” [1].

Hamilton worked unsuccessfully at creating this algebra for over 10 years, and finally he had a breakthrough on October 16, 1843 while walking with his wife, Lady Hamilton. They had been walking along the Royal Canal in Dublin when it occurred to Hamilton that his new algebra would require three rather than two imaginary parts. In order to do this, he could create a new algebraic structure consisting of one real part and three imaginary parts $i, j$ and $k$. For this new structure to work, Hamilton realized that these new imaginary units would have satisfy the following conditions:

$$i^2 = j^2 = k^2 = ijk = -1.$$
Hamilton carved these results on the nearby Broome Bridge. Unfortunately, the carvings no longer remain today [4]. However, his discovery was so significant that every year on October 16th, the Mathematics Department of the National University of Ireland commemorates his discovery by walking to Broome Bridge.

3 The Basic Properties of Quaternions

Algebra

To better understand the discovery of quaternions, it is important to understand them as an algebraic structure. As we indicated in the first section, quaternions are a four dimensional extension of complex numbers with three dimensions being imaginary and the other being real. Therefore, many of the properties of quaternion algebra can be deduced by logically extending the well known properties of complex algebra. Unlike complex numbers, the set of quaternions are not commutative algebra under multiplication. In this section, we will provide the basic algebraic definitions of quaternions.

3.1 Notation and Basic Definition of Quaternions

As we know, the set of complex numbers is defined by

\[ \mathbb{C} = \{a + ib \mid a, b \in \mathbb{R}, i^2 = -1\}. \]

This means that every complex number has one real and one imaginary part and they can be written in the form \(a + ib\) where \(a\) and \(b\) are real numbers and \(i\) is an imaginary unit, \(i^2 = -1\). In addition to \(i\), quaternions are constructed by adding two new imaginary units \(j\) and \(k\) with one real part. Hence, a quaternion \(q\) can be written as a sum of one real part and three imaginary parts

\[ q = q_0 + iq_1 + jq_2 + kq_3 \]

or as four-dimensional vectors

\[ q = (q_0, q_1, q_2, q_3) = (q_0, \vec{q}), \]

where \(q_0\) is the scalar component of \(q\) and \(\vec{q}_0 = (q_1, q_2, q_3)\) constitute vector part and \(i, j, k\) are the standard orthonormal basis for \(\mathbb{R}^3\):

\[ i = (1, 0, 0) \]
\[ j = (0, 1, 0) \]
\[ k = (0, 0, 1). \]
The orthonormal basis components $i$, $j$, $k$, as defined above satisfy the following well-known rules introduced by Hamilton:

\[
\begin{align*}
i^2 &= j^2 = k^2 = ijk = -1 \\
i j &= -j i = k \\
jk &= -k j = i \\
ki &= -i k = j.
\end{align*}
\tag{1}
\]

These relations are often called Hamilton’s Rules. The set of quaternions often are denoted by $\mathbb{H}$ in honor of their discoverer. Hence, the set of quaternions can be written as

\[
\mathbb{H} = \{q = q_0 + iq_1 + jq_2 + kq_3 \mid q_t \in \mathbb{R}, i^2 = j^2 = k^2 = ijk = -1\}
\]

This is clearly an extension of the complex numbers system, where the complex numbers are those quaternions that have $q_2 = q_3 = 0$ and the real numbers are those quaternions that have $q_1 = q_2 = q_3 = 0$.

### 3.2 Equality, Addition, Subtraction and Scalar Multiplication

Due to quaternions are an extension of complex numbers, many properties of them are familiar.
Let \( p = p_0 + ip_1 + jp_2 + kp_3 \) and \( q = q_0 + iq_1 + jq_2 + kq_3 \) be two elements in \( \mathbb{H} \). We say that
\[
p = q \iff p_t = q_t, \quad \forall t \in \{0, 1, 2, 3\}.
\]
Addition and substraction of two quaternions acts component-wise:
\[
p \pm q = (p_0, p_1, p_2, p_3) \pm (q_0, q_1, q_2, q_3) = (p_0 \pm q_0, p_1 \pm q_1, p_2 \pm q_2, p_3 \pm q_3) = (p_0 \pm q_0) + (p_1 \pm q_1)i + (p_2 \pm q_2)j + (p_3 \pm q_3)k.
\]
Every quaternion \( q \) has an additive inverse \( -q \) with components \( -q_i \), \( i = 0, 1, 2, 3 \), \( -q = (-q_0, -q_1, -q_2, -q_3) \).
Quaternions also form a commutative group under addition:
\[
p + q = (p_0, p_1, p_2, p_3) + (q_0, q_1, q_2, q_3) = (p_0 + q_0, p_1 + q_1, p_2 + q_2, p_3 + q_3) = (q_0 + p_0, q_1 + p_1, q_2 + p_2, q_3 + p_3) \quad \text{(Commutativity of } \mathbb{R})
\]
\[
= (q_0, q_1, q_2, q_3) + (p_0, p_1, p_2, p_3) = q + p.
\]
Multiplication of quaternions by a scalar is the same as complex numbers.
For, \( \forall q \in \mathbb{H} \) and \( c \in \mathbb{R} \),
\[
cq = c(q_0, q_1, q_2, q_3) = c(q_0 + iq_1 + jq_2 + kq_3) = cq_0 + icq_1 + jcq_2 + k cq_3 = (cq_0, cq_1, cq_2, cq_3).
\]

3.3 Inner Product, Multiplication and Division of Two Quaternions

Let \( p, q \in \mathbb{H} \). The inner product of two quaternions is similar with three-dimensional vectors, except with an added dimension and defined as
\[
p \cdot q = (p_0, p_1, p_2, p_3) \cdot (q_0, q_1, q_2, q_3)
\]
\[
= p_0q_0 + p_1q_1 + p_2q_2 + p_3q_3
\]
\[
= p_0q_0 + \vec{p} \cdot \vec{q}.
\]
To define the multiplication of two quaternions \( p \) and \( q \) takes a little bit more work. In this thesis, quaternion multiplication will be denoted by the \( \ast \).
operator. Using the distributivity of multiplication over addition, the product becomes

\[ p \star q = (p_0 + ip_1 + jp_2 + kp_3) \star (q_0 + iq_1 + jq_2 + kq_3) \]
\[ = p_0q_0 + p_0q_1 + jp_0q_2 + kp_0q_3 \]
\[ + ip_1q_0 + i^2p_1q_1 + ijp_1q_2 + ikp_1q_3 \]
\[ + jp_2q_0 + jip_2q_1 + j^2p_2q_2 + jkp_2q_3 \]
\[ + kp_3q_0 + kip_3q_1 + kjp_3q_2 + k^2p_3q_3. \]  
(2)

It is quite long to understand what is going on. However, Eq. (2) can be simplified by using Hamilton's Rules from Eq. (1). But, we must be careful simplifying this step because multiplication is commutative for real numbers but not for imaginary elements \((ij \neq ji)\). Using Eq. (1), we can rewrite the multiplication again as,

\[ p \star q = p_0 + i(p_1q_0 + p_0q_1 + p_2q_2 + p_3q_3) \]
\[ + i(p_1q_0 - p_1q_1 + k(p_1q_2 - jp_1q_3) \]
\[ + j(p_2q_0 - kp_2q_1 - p_2q_2 + i_2p_2q_3 \]
\[ + k(p_3q_0 + j(p_3q_1 - ip_3q_2 - k(p_3q_3. \]

We regroup the terms according to the imaginary units, and we obtain

\[ p \star q = p_0q_0 - (p_1q_1 + p_2q_2 + p_3q_3) + i(p_1q_0 + p_0q_1 + p_2q_2 + p_3q_3) \]
\[ + j(p_2q_0 + p_0q_2 + p_3q_1 - p_1q_3) + k(p_3q_0 + p_0q_3 - p_1q_2 - p_2q_1) \]

We may also write multiplication rule in matrix form [5], as

\[ p \star q = \begin{bmatrix} p_0 & 0 & 0 & 0 \\
p_1 & p_0 & 0 & 0 \\
p_2 & 0 & p_0 & 0 \\
p_3 & 0 & 0 & p_0 \end{bmatrix} \begin{bmatrix} q_0 \\
q_1 \\
q_2 \\
q_3 \end{bmatrix} \]
\[ = \begin{bmatrix} p_0q_0 - p_1q_1 - p_2q_2 - p_3q_3 \\
p_1q_0 + p_0q_1 + p_2q_3 - p_3q_2 \\
p_2q_0 + p_0q_2 + p_3q_1 - p_1q_3 \\
p_3q_0 + p_0q_3 + p_1q_2 - p_2q_1 \end{bmatrix} \]
\[ = (p_0q_0 - \vec{p} \cdot \vec{q}, p_0q_1 + p_0q_2 + p_0q_3 + \vec{p} \times \vec{q}). \]  
(3)

From Eq. (3), we can see that quaternions are not commutative under multiplication. Because, the cross product is used to define our multiplication and
as we know, the cross product is not commutative. It gives different results depending on order of the vectors \( \vec{p} \times \vec{q} \neq \vec{q} \times \vec{p} \). Also, we see that the product of two quaternions is still a quaternion with scalar part \( p_0 q_0 - \vec{p} \cdot \vec{q} \) and vector part \( p_0 \vec{q} + q_0 \vec{p} + \vec{p} \times \vec{q} \). This means that the set of quaternions is closed under multiplication.

Quaternion multiplication is also associative and distributes over addition:

\[
\begin{align*}
(p \ast q) \ast r &= p \ast (q \ast r) \\
(p + q) \ast r &= p \ast r + q \ast r \\
p \ast (q + r) &= p \ast q + p \ast r.
\end{align*}
\]

The multiplicative identity quaternion has real part 1 and vector part 0, so

\[ 1 = (1, \vec{0}) = (1, 0, 0, 0) = 1 + i0 + j0 + k0. \]

This can be directly checked by using Eq. (3). For \( p = (p_0, \vec{p}) \in \mathbb{H} \),

\[
\begin{align*}
(p_0, \vec{p}) \ast (1, \vec{0}) &= (p_0 - \vec{p} \cdot \vec{0}, p_0\vec{0} + 1\vec{p} + \vec{p} \times \vec{0}) = (p_0, \vec{p}), \\
(1, \vec{0}) \ast (p_0, \vec{p}) &= (p_0 - \vec{0} \cdot \vec{p}, 1\vec{p} + p_0\vec{0} + \vec{0} \times \vec{p}) = (p_0, \vec{p}).
\end{align*}
\]

Quaternion division is non-commutative as well and is defined by the (order-dependent) relations

\[
\frac{u}{v} = u \ast \frac{\vec{v}}{|v|^2}
\]

and

\[
\frac{v}{u} = \vec{v} \ast \frac{u}{|v|^2}.
\]

Here, \( \vec{v} \) is denoted as the conjugate of a quaternion and \( |v| \), denoted the norm of a quaternion which will be described in the next section.

### 3.4 The Complex Conjugate, Norm, Inverse and Unit Quaternions

As in the case of complex numbers, we can define the conjugate of a quaternion. Let \( q \in \mathbb{H} \). The complex conjugate of \( q \), denoted \( \bar{q} \), is defined as

\[
\bar{q} = (q_0, -\vec{q}) = (q_0, -q_1, -q_2, -q_3) = q_0 - iq_1 - jq_2 - kq_3.
\]
From the above definition, we immediately have

\[
\overline{(q)} = (q_0, -\vec{q}) = (q_0, -(-\vec{q})) = (q_0, \vec{q}) = q
\]
\[
q + \overline{q} = (q_0, \vec{q}) + (q_0, -\vec{q}) = (2q_0, 0) = 2q_0.
\]

Also, given two quaternions \( p \) and \( q \), we can easily verify that

\[
\overline{(p \ast q)} = \overline{q} \ast \overline{p}.
\]

Furthermore, the Eq. (4) allows us to define the quaternion norm \( |q| \) as

\[
|q| = \sqrt{\overline{q} \ast q}.
\]

It is important to note that

\[
|q|^2 = \overline{q} \ast q
= (q_0, -\vec{q}) \ast (q_0, \vec{q})
= (q_0q_0 - (-\vec{q}) \cdot \vec{q}, q_0\vec{q} + (-\vec{q})q_0 + (-\vec{q}) \times \vec{q}) 
= q_0^2 + \vec{q} \cdot \vec{q}
= q_0^2 + q_1^2 + q_2^2 + q_3^2
= q \ast \overline{q}.
\]

Using Eq. (5), it can be shown that the quaternion norm is multiplicative. This means that the norm of the product of two quaternions \( p \) and \( q \) is the product of the individual norms. We can check this fact as follows [5]:

\[
|p \ast q|^2 = (p_0q_0 - \vec{p} \cdot \vec{q})^2
+((p_0\vec{q} + q_0\vec{p}) + \vec{p} \times \vec{q}) \cdot ((p_0\vec{q} + q_0\vec{p}) + \vec{p} \times \vec{q})
= (p_0q_0)^2 - 2p_0q_0\vec{p} \cdot \vec{q} + (\vec{p} \cdot \vec{q})^2
+((p_0\vec{q} + q_0\vec{p})^2 + 2(p_0\vec{q} + q_0\vec{p}) \cdot (\vec{p} \times \vec{q}) + (\vec{p} \times \vec{q}) \cdot (\vec{p} \times \vec{q})
= (p_0q_0)^2 - 2p_0q_0\vec{p} \cdot \vec{q} + (\vec{p} \cdot \vec{q})^2
+((p_0\vec{q} + q_0\vec{p})^2 + (\vec{p} \times \vec{q}) \cdot (\vec{p} \times \vec{q})
+ (2(p_0\vec{q} + q_0\vec{p}) \cdot (\vec{p} \times \vec{q}) = 0)
= (p_0q_0)^2 - 2p_0q_0\vec{p} \cdot \vec{q} + (\vec{p} \cdot \vec{q})^2
+((p_0\vec{q} + q_0\vec{p})^2 + (\vec{p} \cdot \vec{q})^2 - (\vec{p} \cdot \vec{q})^2
= (p_0q_0)^2 + (q_0)^2(\vec{p})^2 + (p_0)^2(\vec{q})^2 + (\vec{p})^2(\vec{q})^2
= (|p_0|^2 + |\vec{p}|^2)(|q_0|^2 + |\vec{q}|^2)
= |p|^2|q|^2.
\]

In the above equation, we wrote \((\vec{p})^2(\vec{q})^2 - (\vec{p} \cdot \vec{q})^2\) instead of \((\vec{p} \times \vec{q}) \cdot (\vec{p} \times \vec{q})\). Because of the tedious calculation, we will not give the proof of this equality.
Since we have defined the conjugate and the norm, we can use them to describe the inverse of a quaternion. But, before describing the inverse, it is better to define a unit quaternion. A quaternion is called a unit quaternion (also called a normalized quaternion) if its norm is 1. We will use $\mathbb{H}_1$ to denote the set of unit quaternions, i.e. the set of all $q \in \mathbb{H}$ such that

$$|q| = q_0^2 + \vec{q} \cdot \vec{q} = q_0^2 + q_1^2 + q_2^2 + q_3^2 = 1.$$ 

Also note that any quaternion $q$ can be normalized by dividing it by its norm, to obtain a unit quaternion. If we donate a unit quaternion as $q_u$, then

$$q_u = \frac{q}{|q|}, \quad |q| \neq 0.$$ 

We can now show that every nonzero quaternion $q$ has a multiplicative inverse. For a non zero quaternion $q$, the inverse of $q$, denoted $q^{-1}$, is defined as

$$q^{-1} = \frac{\bar{q}}{|q|^2}, \quad |q| \neq 0.$$ 

We can easily verify that $q^{-1} \ast q = q \ast q^{-1} = 1$. In this case, if $q$ is a unit quaternion, then the inverse is its conjugate $\bar{q}$. ($q^{-1} = \bar{q}$)

The set of unit quaternions forms a unit sphere in four-dimensional space. We shall later see that the set of unit quaternions play an important part in relation to general rotations.

3.5 Alternative Ways of Writing Quaternions

As we referred at the beginning of this section, quaternions are represented simply as a four vector, a list of four variables $(q = (q_0, q_1, q_2, q_3) = (q_0, \vec{q}))$ that obeys certain rules. (Hamilton’s Rules, see Section 3.1)

There are several different ways that quaternions can be written and it is helpful to know them all as each for is useful. In this section, we will denote two different ways in addition to Hamilton’s definition to write quaternions [5].
Pauli Matrices
The Pauli matrices are a set of three $2 \times 2$ matrices and they are usually indicated by the Greek letter $\sigma$. The Pauli matrices are written by convention as

$$
\sigma_1 = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}, \\
\sigma_2 = \begin{bmatrix} 0 & -i \\ i & 0 \end{bmatrix}, \\
\sigma_3 = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix},
$$

where $i$ is the usual imaginary number with $i^2 = -1$. We can immediately verify that the Pauli matrices obey the following relationships:

$$
I_2 = (\sigma_1)^2 = (\sigma_2)^2 = (\sigma_3)^2, \\
\sigma_1 \sigma_2 = i \sigma_3, \\
\sigma_2 \sigma_3 = i \sigma_1, \\
\sigma_3 \sigma_1 = i \sigma_2, \\
-i \sigma_1 \sigma_2 \sigma_3 = I_2.
$$

Here, as we know, $I_2$ is the $2 \times 2$ identity matrix, $I_2 = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$.

Now any quaternion can be written

$$
q = q_0 I_2 - i \sigma_1 q_1 - i \sigma_2 q_2 - i \sigma_3 q_3 \\
= \begin{bmatrix} q_0 - iq_3 & -iq_1 - q_2 \\ -iq_1 + q_2 & q_0 + iq_3 \end{bmatrix} \\
= q_0 I_2 - i \vec{\sigma} \cdot \vec{q},
$$

where in the last expression, we can see that $-i \vec{\sigma} = -i(\sigma_1, \sigma_2, \sigma_3)$ and $\vec{q} = (q_1, q_2, q_3)$ as well.

Other Matrix Forms
Another way of writing quaternions is to use $4 \times 4$ orthogonal matrix forms. These matrices are defined as

$$
\Lambda_1 = \begin{bmatrix} 0 & -1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 \\ 0 & 0 & +1 & 0 \end{bmatrix},
$$
\[ \Lambda_2 = \begin{bmatrix} 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & +1 \\ +1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \end{bmatrix}, \]

\[ \Lambda_3 = \begin{bmatrix} 0 & 0 & 0 & -1 \\ 0 & 0 & -1 & 0 \\ 0 & +1 & 0 & 0 \\ +1 & 0 & 0 & 0 \end{bmatrix}, \]

where

\[-I_4 = \Lambda_1^2 = \Lambda_2^2 = \Lambda_3^2,\]

\[\Lambda_1 \Lambda_2 = \Lambda_3,\]

\[\Lambda_2 \Lambda_3 = \Lambda_1,\]

\[\Lambda_3 \Lambda_1 = \Lambda_2.\]

Then our typical quaternion can be written (using these matrices) as

\[ q = q_0 I_4 + \bar{\Lambda} \cdot \bar{q} \]

\[= \begin{bmatrix} q_0 & -q_1 & -q_2 & -q_3 \\ q_1 & q_0 & -q_3 & q_2 \\ q_2 & q_3 & q_0 & -q_1 \\ q_3 & -q_2 & q_1 & q_0 \end{bmatrix}. \tag{6} \]

In this representation, the fourth power of the norm of a quaternion is the determinant of the corresponding matrix. Moreover, if \( q \) is a unit quaternion, then the conjugate (or the inverse) of this quaternion corresponds to the transpose \( (T) \) of this matrix [6].

If we denote the matrix in Eq. (6) with \( A \), then we can verify that

\[ \det A = |q|^4 \]

\[ A^T = \bar{q} = q^{-1}, \text{ if } q \text{ is a unit quaternion.} \]

It is really tedious to prove these expressions so we will not show them.
4 Rotation Representations

There are several different ways to represent rotations in 3D vector space ($\mathbb{R}^3$). In this section, we will study the three most important representation methods for rotation in $\mathbb{R}^3$. These are rotation matrices, Euler angles and quaternions.

The aim of this section is to reach an implementation of general rotation in $\mathbb{R}^3$ with each way. A comparison of what are the advantages and disadvantages of these methods is given section 5.

In section 4.1, rotation group and rotation matrices are described. Euler angles and their matrix representation are given in section 4.2. Finally, section 4.3 describes how quaternions can be used to represent rotation in $\mathbb{R}^3$.

4.1 Rotation Group and Rotation Matrices

The rotation group is the group which represents all orthogonal rotations in three-dimensional Euclidean space ($\mathbb{R}^3$). The group of rotations in $\mathbb{R}^3$ is usually denoted as $SO(3)$, which stands for the group of special orthogonal $3 \times 3$ matrices. Recall that an orthogonal matrix consists of orthogonal column vectors which have unit magnitude. In other words, the column vectors form an orthonormal basis of $\mathbb{R}^3$. An element of $SO(3)$ can be represented by a rotation matrix.

A rotation matrix in $\mathbb{R}^3$ is a $3 \times 3$ matrix representing a rotation about a particular axis through a defined angle. A rotation matrix $R \in SO(3)$ is categorized as orthogonal matrix with determinant 1:

$$R^T = R^{-1},$$
$$\det R = 1.$$  

Also, $R \in SO(3)$ will transform a column vector $\vec{x} \in \mathbb{R}^3$ to a new column vector

$$\vec{y} = R\vec{x}$$

by rotating it about an axis with a rotation angle and preserving its magnitude which means that $|\vec{x}| = |\vec{y}|$. 

12
Rotations about the $x$-axis by the angle $\alpha$, $y$-axis by the angle $\beta$ and $z$-axis by the angle $\gamma$, respectively written as $R_x(\alpha)$, $R_y(\beta)$ and $R_z(\gamma)$ are

\[
R_x(\alpha) = \begin{bmatrix}
1 & 0 & 0 \\
0 & \cos \alpha & -\sin \alpha \\
0 & \sin \alpha & \cos \alpha
\end{bmatrix}
\] (7)

\[
R_y(\beta) = \begin{bmatrix}
\cos \beta & 0 & \sin \beta \\
0 & 1 & 0 \\
-\sin \beta & 0 & \cos \beta
\end{bmatrix}
\] (8)

\[
R_z(\gamma) = \begin{bmatrix}
\cos \gamma & -\sin \gamma & 0 \\
\sin \gamma & \cos \gamma & 0 \\
0 & 0 & 1
\end{bmatrix}
\] (9)

where rotation angles $\alpha$, $\beta$, $\gamma$ about coordinate axis ($x, y,$ and $z$) are called Euler angles. Also, it is important to point out that each of these rotations are counter-clockwise and the coordinate system is right-handed. Because, the fundamental system is arbitrary; it can be right-handed or left-handed. In this thesis, we use right-handed coordinate system (see Figure 4.1) in all functions implemented during this work.

![Figure 4.1: The right-handed coordinate system [7]](image)
4.2 Euler Angles and Rotation Sequences

A common way to represent a rotation in three-dimensional Euclidean space is to use Euler angle sequences. Leonard Euler showed that any rotation can be completed successfully by a sequence of three rotations about the coordinate axis. An Euler angle sequence is a rotation matrix that is completely determined by three parameters, called Euler angles, $\alpha$, $\beta$, $\gamma$ with each corresponding to a rotation matrix which we defined in section 4.1. (see Eq. 7-9)

Since there are three axes, there are total 27 possible rotation sequences. Each rotation sequence gives different results and different rotation sequences can be used in different applications. For example, one of the most common conventions, used in the field of engineering and classical mechanics, is called the “zxz-convention” which is rotation first about the $z$-axis, then the $x$-axis, and then again the $z$-axis. Another common convention is the “xyz-convention” which is used in the field of aerospace engineering and computer graphics [8]. In this thesis, we will use “xyz-convention”, which consists of a rotation about the $x$-axis, then the $y$-axis and the last the $z$-axis.

If we consider the rotation matrices in Eq. (7-9), we can write our rotation sequence (for $xyz$-convention) in the form of a matrix as below:

$$R(\alpha, \beta, \gamma) = R_z(\gamma)R_y(\beta)R_x(\alpha)$$

$$= \begin{bmatrix}
\cos \beta \cos \gamma & -\cos \alpha \sin \gamma + \sin \alpha \sin \beta \cos \gamma & \sin \alpha \sin \gamma + \cos \alpha \sin \beta \cos \gamma \\
n\sin \beta \cos \gamma & \cos \alpha \cos \gamma + \sin \alpha \sin \beta \sin \gamma & -\sin \alpha \cos \gamma + \cos \alpha \sin \beta \sin \gamma \\
-n\sin \beta & \sin \alpha \cos \beta & \cos \alpha \cos \beta
\end{bmatrix}$$

(10)

Now, we have a rotation matrix that will allow us to define any rotation in $\mathbb{R}^3$ using the $xyz$-convention with Euler angles. Let’s consider a vector, $\vec{v} \in \mathbb{R}^3$. If we rotate this vector by using Euler angle sequence $R(\alpha, \beta, \gamma)$, the vector $\vec{v}$ produces a new vector $\vec{v}' \in \mathbb{R}^3$, according to

$$\vec{v}' = R(\alpha, \beta, \gamma)\vec{v} = R_z(\gamma)R_y(\beta)R_x(\alpha)\vec{v}.$$ 

But, this method causes limitations in representing rotation of an object. Because, in every Euler angle sequence, there is at least one point where we lose a degree of freedom. This loss of degree of freedom is known as gimbal lock [9] which is explained within the next few sections.

4.3 Rotation with Quaternions

In this section, we show how quaternions are used to rotate vectors about an arbitrary axis. Quaternions are a very efficient way to represent rotations
because they define an angle and rotation axis. Before starting to show how they can be used, it is better to define a pure quaternion.

A pure quaternion is defined as a quaternion whose scalar part is zero. In other words, a quaternion of the form \((0, \vec{q}) = (0, q_1, q_2, q_3) = 0 + iq_1 + jq_2 + kq_3\) is called pure. Let \(\mathbb{H}_0\) denote the set of pure quaternions. If we think about linear algebra, we will recognize that \(\mathbb{H}_0\) is a 3-dimensional real vector space (\(\mathbb{R}^3\)). In a sense, a standard 3D vector are stored in a pure quaternion \((\vec{q} = (q_1, q_2, q_3) = (0, \vec{q}))\).

Now, we can define quaternion rotation operator. Let \(q \in \mathbb{H}_1\) and \(v \in \mathbb{H}_0\). Using the unit quaternion \(q\), we define an operator on vector \(\vec{v} \in \mathbb{R}^3\):

\[
R_q(\vec{v}) = q \ast \vec{v} \ast q^{-1} = q \ast \vec{v} \ast \bar{q}, \quad (q^{-1} = \bar{q})
\]

\[
= (q_0, \vec{q}) \ast (0, \vec{v}) \ast (q_0, -\vec{q})
\]

\[
= (0, (q_0^2 - |\vec{q}|^2)\vec{v} + 2(\vec{q} \cdot \vec{v})\bar{q} + 2q_0(\vec{q} \times \vec{v})).
\]

Note that to obtain \(R_q(\vec{v})\) as in Eq. (10), we used a formula [10] which is known as a vector triple product in linear algebra and is given as below:

**Lemma:** Let \(\vec{u} = (u_1, u_2, u_3), \vec{v} = (v_1, v_2, v_3)\) and \(\vec{w} = (w_1, w_2, w_3) \in \mathbb{R}^3\), then the vector triple product is defined by:

\[
\vec{u} \times (\vec{v} \times \vec{w}) = (\vec{u} \cdot \vec{w})\vec{v} - (\vec{u} \cdot \vec{v})\vec{w}.
\]

Furthermore, the Eq. (10) shows that the quaternion rotation operator \(R_q(\vec{v})\) is a pure quaternion. Therefore, \(R_q(\vec{v})\) defines a function \(\mathbb{R}^3 \rightarrow \mathbb{R}^3\). It is also simple to demonstrate that the quaternion rotation operator \(R_q(\vec{v})\) is a linear operator and preserves the length of a vector [11].

### 4.4 Geometric Interpretation of Unit Quaternions

As we stated in the last section, unit quaternions are used to rotate vectors in \(\mathbb{R}^3\). Now, we will represent unit quaternions in an alternative way. Let us consider a unit quaternion, \(q = (q_0, \vec{q}) = q_0 + \vec{q}\). Since \(q\) is a unit quaternion,

\[
\begin{cases}
|q|^2 = 1, \\
q_0^2 + \vec{q} \cdot \vec{q} = 1, \\
q_0^2 + |\vec{q}|^2 = 1.
\end{cases}
\]

By using the well-known trigonometric formula \(\cos^2 \theta + \sin^2 \theta = 1\) with Eq. (11), we obtain

\[q_0^2 + |\vec{q}|^2 = \cos^2 \theta + \sin^2 \theta.\]
This implies that there exists \( \theta \) such that \(-\pi < \theta < \pi\) and

\[
\begin{cases}
q_0^2 = \cos^2 \theta, \\
|\vec{q}|^2 = \sin^2 \theta
\end{cases}
\]  
(12)

And, therefore, we define a vector \( \vec{u} \) as

\[
\begin{cases}
\vec{u} = \frac{\vec{q}}{|\vec{q}|} = \frac{\vec{q}}{\sin \theta} \\
\dot{\vec{q}} = \vec{u} \sin \theta.
\end{cases}
\]  
(13)

Combining Eq. (12) with Eq. (13), any unit quaternion can be written in terms of an angle \( \theta \) and a unit vector \( \vec{u} = \vec{q}/|\vec{q}| \) as

\[
q = \cos \theta + \vec{u} \sin \theta.
\]  
(14)

As a conclusion, for any unit quaternion

\[
q = q_0 + \vec{q} = \cos \theta + \vec{u} \sin \theta
\]

and for any vector \( \vec{v} \in \mathbb{R}^3 \), the action of the operator

\[
R_q(\vec{v}) = q \ast \vec{v} \ast \bar{q}
\]
on \( \vec{v} \) may be interpreted geometrically as a rotation of the vector \( \vec{v} \) through an angle \( 2\theta \) about \( \vec{q} \) as the axis of rotation \([12]\).

### 4.5 Gimbal Lock

Although Euler Angles are the traditional way to represent rotations in three-dimensional space, they cause a few problems. One of the problem that occurs when using Euler angles is the possibility of mathematical singularities. This problem is called gimbal lock\(^1\).

Gimbal lock is a phenomenon where one of the rotation axis realign with the other axis and causes loss of one degree of freedom. Starting from the initial configuration (Figure 4.1), we observe that each of the configurations a through c corresponds to rotations about a single axis.

---

\(^1\)We recommend http://www.youtube.com/watch?v=zc8b2Jo7mno for an illustration of the phenomenon.
For example, for rotation defined in the \textit{xyz-convention}, if we rotate an object by 90 degrees about the \textit{y-axis} (axis 2 in figure 4.2) and then the \textit{x-axis} (axis 1) and \textit{z-axis} (axis 3) line up, so that we will effectively lose one of our degrees of freedom.
Figure 4.2 is a visual example to show gimbal lock. Within the next sections, we will explain gimbal lock mathematically and give a concrete example to show the difference between Euler angles and quaternions.

5 Comparison Between Euler Angles and Quaternions

In the previous section, we introduced two main rotational methods: One of them is rotation defined by Euler angles represented by rotation matrices and the other method is rotation defined by quaternions. In this section, we will describe advantages and disadvantages of these methods.

Traditionally, the most common method that is used to parameterize is to use Euler angles, because, Euler angle sequences are relatively easy to understand. Although they are easy to understand for users, they are inadequate to represent all rotations. Also, they are not so easy to represent mathematically. For instance, if we want to rotate an object 30 degrees about rotation axes \((x, y, z)\) given by the vector \((1, 2, 3)\), it is quite tedious to find the corresponding Euler angles sequences about the three basis axes.

The representation of rotations by quaternions has several advantages over the representation by Euler angles. The parametrization of rotations using quaternions involve only the angle and the axis (vector) of rotation, while Euler angles define a rotation as a composition of three independent rotations about coordinate axes.

Besides, as discussed earlier, gimbal lock can cause problems with Euler angle system. Even different formulations of the Euler angles in the rotation matrices from Eq. (7)-(9) does not remove this singularity. Now, we will give an example to better understand the gimbal lock difference between Euler angles and quaternions. First, occurrence of singularity using Euler angles is demonstrated using \(xyz\) rotation sequence. Later, quaternion is applied in similar sequence, to show that singularity can be avoided when quaternion is used.

Let us first consider rotation about \(x\)-axis by 0 degree, followed by rotation about \(y\)-axis by 90 degrees and finally rotation about \(z\)-axis by an angle of \(\gamma\). In this case, resulting of Euler angle sequence can be obtained by substituting \(\alpha = 0\), \(\beta = 90\) degrees and \(\gamma = \gamma\) degree into Eq. (10);

\[
R(0, 90, \gamma) = \begin{bmatrix}
0 & -\sin \gamma & \cos \gamma \\
0 & \cos \gamma & \sin \gamma \\
-1 & 0 & 0
\end{bmatrix}.
\]
Now, consider rotation about $x$-axis by an angle of $\alpha$, followed by rotation about $y$-axis by 90 degrees and finally rotation about $z$-axis by an angle of 0 degree. Resulting of Euler angle sequence for this case can be obtained by substituting $\alpha = -\alpha$, $\beta = 90$ degree and $\gamma = 0$ degree into Eq. (10);

$$R(-\alpha, 90, 0) = \begin{bmatrix}
0 & -\sin \alpha & \cos \alpha \\
0 & \cos \alpha & \sin \alpha \\
-1 & 0 & 0
\end{bmatrix}.$$ 

It can be seen that both cases give similar Euler angle sequence. Because, rotation about $y$-axis by 90 degree causes the $z$ and $x$ axes to align with each other. Therefore, singularity (gimbal lock) occurs. Now, we will show that this can be avoided by using quaternions [14].

Let $q_x, q_y, q_z \in \mathbb{H}_1$ and $L_{q_x}({\vec{v}}), L_{q_y}({\vec{v}})$ and $L_{q_z}({\vec{v}})$ be the rotations around the $x$-axis, $y$-axis and $z$-axis respectively. As we mentioned in section 4.4, any unit quaternion can be written in terms of an angle and a unit vector. If we describe the angles $\alpha$ for rotation around the $x$-axis, $\beta$ for rotation around the $y$-axis and $\gamma$ for rotation around the $z$-axis and $i, j, k$ as unit vectors, then by Eq. (14),

$$q_x = \cos \frac{\alpha}{2} + i \sin \frac{\alpha}{2}$$
$$q_y = \cos \frac{\beta}{2} + j \sin \frac{\beta}{2}$$
$$q_z = \cos \frac{\gamma}{2} + k \sin \frac{\gamma}{2}$$

It is important to note that we are defining our $q_x$ (similarly, $q_y$ and $q_z$) with $\frac{\alpha}{2}$, (or $\frac{\beta}{2}$ and $\frac{\gamma}{2}$) inside the cosine and sine functions. So, $L_{q_x}$, (or $L_{q_y}$ and $L_{q_z}$) rotates $\vec{v}$ about the $x$-axis, (or $y$ and $z$-axes) at an angle of $\alpha$, (or $\beta$ and $\gamma$) rather than $2\alpha$, (or $2\beta$ and $2\gamma$). Now, let us define $L_q$ as the rotation about first the $x$-axis, then $y$-axis and last the $z$-axis. So,

$$L_q(\vec{v}) = L_{q_z}(L_{q_y}(L_{q_x}(\vec{v}))) = q_z \ast q_y \ast q_x \vec{v}q_x^{-1} \ast q_y^{-1} \ast q_z^{-1}$$

From the above equation, it is easy to see that $q = q_z \ast q_y \ast q_x$. Using the multiplication rule from Eq. (10), we multiply $q_z \ast q_y \ast q_x$ to find the four components $(q_0, q_1, q_2, q_3)$ of $q$. After the tedious calculations, we obtain that

$$\begin{cases}
q_0 = \cos \frac{\gamma}{2} \cos \frac{\beta}{2} \cos \frac{\alpha}{2} + \sin \frac{\gamma}{2} \sin \frac{\beta}{2} \sin \frac{\alpha}{2} \\
q_1 = \cos \frac{\gamma}{2} \cos \frac{\beta}{2} \sin \frac{\alpha}{2} - \sin \frac{\gamma}{2} \sin \frac{\beta}{2} \cos \frac{\alpha}{2} \\
q_2 = \cos \frac{\gamma}{2} \sin \frac{\beta}{2} \cos \frac{\alpha}{2} + \sin \frac{\gamma}{2} \cos \frac{\beta}{2} \sin \frac{\alpha}{2} \\
q_3 = \sin \frac{\gamma}{2} \cos \frac{\beta}{2} \cos \frac{\alpha}{2} - \cos \frac{\gamma}{2} \sin \frac{\beta}{2} \sin \frac{\alpha}{2}
\end{cases} \quad (15)
As we did before, recreating the first case by substituting $\alpha = 0$, $\beta = 90$ degrees and $\gamma = \gamma$ degree into Eq. (15) gives

\[
q_0 = \frac{1}{\sqrt{2}} \cos \frac{\gamma}{2} \\
q_1 = \frac{-1}{\sqrt{2}} \sin \frac{\gamma}{2} \\
q_2 = \frac{1}{\sqrt{2}} \cos \frac{\gamma}{2} \\
q_3 = \frac{1}{\sqrt{2}} \sin \frac{\gamma}{2}
\]

Recreating the second case by substituting $\alpha = -\alpha$, $\beta = 90$ degrees and $\gamma = 0$ degree into Eq. (15) gives

\[
q_0 = \frac{1}{\sqrt{2}} \cos \frac{\alpha}{2} \\
q_1 = \frac{1}{\sqrt{2}} \sin \frac{\alpha}{2} \\
q_2 = \frac{1}{\sqrt{2}} \cos \frac{\alpha}{2} \\
q_3 = \frac{-1}{\sqrt{2}} \sin \frac{\alpha}{2}
\]

Now, it is clear that both cases give different answers. Thereby, by using quaternions, we can avoid singularity [14].

As we showed above, quaternions can be used to represent rotations without singularity. Also, quaternions are used in many applications such as molecular modeling [15], gyroscopic motion, navigation and guidance [16], orbital mechanics of satellites [12], robotics [17], signal processing [18], and flight simulators [19].

Another common use for quaternions is in the development of computer games, graphics and 3D virtual worlds [20]. In recent years, graphics and game programmers discovered the true potential of quaternions and started using them for describing rotations [21]. When we compared to another common method known as Euler angle sequences, quaternions can be easily used in these applications to obtain smooth rotation by interpolating between orientations, known as spherical linear interpolation (SLERP). Now, we will introduce this method, which is increasingly used by animators.
5.1 Spherical Linear Interpolation

In several applications in computer graphics it is important to interpolate rotations. For example, a position of an articulate body is usually determined by the rotations of each joint and in animations, they are created by interpolating between these rotations.

In Euclidean space, the easiest way to interpolate between two points is the use of linear interpolation (LERP). LERP is a basic geometric formula; given the starting and ending points $q_0$ and $q_1$ and the interpolation parameter $t \in [0, 1]$, $\text{LERP}(q_0, q_1, t)$ gives for each $t$ a point along the straight line connecting them, by the rule (see Figure 4.3)

$$\text{LERP}(q_0, q_1; t) = (1 - t)q_0 + tq_1.$$

(16)

However, a straight line is not useful to animations since a rotating joint is supposed to move along a smooth curve. For this, the most efficient way of interpolating rotations is by means of quaternions. For this, one can use spherical linear interpolation (SLERP). SLERP spans the shortest great arc between two points on the unit quaternion sphere. This interpolation can be stated in the quaternion algebra as follows [21]:

For given $q_0, q_1 \in \mathbb{H}_1$ and $t \in [0, 1]$;

$$\text{Slerp}(q_0, q_1; t) = q_0 \star (q_0^{-1} \star q_1)^t.$$

(17)

Also note that the following four functions are equivalent expressions for
spherical linear interpolation [22]:

\[
Slerp(q_0, q_1; t) = q_0 \ast (q_0^{-1} \ast q_1)^t
= q_1 \ast (q_1^{-1} \ast q_0)^{1-t}
= (q_0 \ast q_1^{-1})^{1-t} \ast q_1
= (q_1 \ast q_0^{-1})^t \ast q_0.
\]

Alternatively, Eq. (17) can be written as

\[
Slerp(q_0, q_1; t) = q_0 \frac{\sin(1-t)\phi}{\sin \phi} + q_1 \frac{\sin t\phi}{\sin \phi},
\]

where \( \phi \) is the angle between \( q_0 \) and \( q_1 \) and \( \cos \phi = q_0 \cdot q_1 \) [5]. From the above equation, the symmetry can be seen in the fact that \( Slerp(q_0, q_1; t) = Slerp(q_1, q_0; 1-t) \). Also, in the limit as \( \phi \to 0 \), this formula reduces to the formula for linear interpolation (Eq. (16)).

As a conclusion, the interpolation curve for Slerp (figure 4.4) forms a great arc on the quaternion unit sphere. Not only great arc, Slerp follows the shortest great arc. Thus, Slerp gives the shortest possible interpolation path between the two quaternions on the unit sphere. So, Slerp is the optimal interpolation curve between two rotations.

Figure 4.4: An illustration in the plane of the difference between Lerp and Slerp. (a) The interpolation covers the angle \( \nu \) in three steps. (b) Lerp: The secant across is split in four equal pieces. The corresponding angles are shown. (c) Slerp: The angle is split in four equal angles. [22]

Notice that SLERP only allows interpolating between two rotations. If there are only two points between to interpolate, we simply calculate \( Slerp(q_0, q_1; t) \) (Eq. (17) and (18)). But, what if we want to interpolate not two, but several points on a sphere i.e three \( (q_{n-1}, q_n, q_{n+1}) \)? We could then interpolate first \( q_{n-1} \) and \( q_n \) and afterwards (separately) \( q_n \) and \( q_{n+1} \). But this method can have some problems. The algorithm can not work properly. To avoid
this, Ken Shoemake suggests the use of the parameterized Bézier curves [21],[22]. However, we will not address this issue in this work.

6 Octonions

It is clear from the stories of William R. Hamilton that quaternions were discovered because he persisted in asking the question “What generalizes the complex numbers”. Therefore, it is appropriate to continue our study of quaternions by asking the question “What do we discover when we attempt to generalize quaternions?” Although the technical answer to this question is complex but it was quickly found that by giving certain mathematical conditions, there is exactly one further generalization which is called octonions. In this section, we present a general overview about the octonions.

6.1 Constructing the Octonions

The octonions were discovered in 1843 by John T. Graves. The story of octonions is nearly as interesting as that of quaternions. Soon after Hamilton discovered quaternions, he sent a letter describing them to his friend John T. Graves. On 26 December 1843, Graves wrote to Hamilton to tell him that he has successfully generalized quaternions to the “octaves”, an 8-dimensional (8D) algebra [23]. Unfortunately for Graves, Hamilton put off assisting him in publishing his results and octonions were discovered independently and published in 1845 by Arthur Cayley.

The octonions are a division algebra\(^2\) [5] over the real numbers. There are four possible algebraic number systems; real numbers (\(\mathbb{R}\)), complex numbers (\(\mathbb{C}\)), quaternions (\(\mathbb{H}\)) and the octonions, usually represented by \(\mathbb{O}\). The octonions are the largest such algebra with eight components:

\[ u = (u_0, u_1, u_2, u_3, u_4, u_5, u_6, u_7) = (u_0, \vec{u}) \]

or every octonion \(u\) can be written as a linear combination of the unit octonions [24]:

\[ u = u_0e_0 + u_1e_1 + u_2e_2 + u_3e_3 + u_4e_4 + u_5e_5 + u_6e_6 + u_7e_7. \]

Thus, we can write the set of octonions as a vector space:

\[ \mathbb{O} = \{ u_0 + \sum_{i=1}^{7} u_i e_i : a_0, ..., a_7 \in \mathbb{R} \}. \]  

\(^2\)Every non-zero element has a multiplicative inverse.
Due to octonions are an extension of quaternions, their many properties are so similar with quaternions. Addition and subtraction of octonions is done by adding and subtracting corresponding terms and hence their coefficients, like quaternions. Multiplication is more complex. Multiplication is distributive over addition, so the product of two octonions can be calculated by summing the product of all the terms. The product of each term can be given by multiplication of the coefficients and a multiplication table of the unit octonions is given below [25]:

\[
\begin{array}{cccccccc}
\ast & e_0 & e_1 & e_2 & e_3 & e_4 & e_5 & e_6 & e_7 \\
\hline
e_0 & e_0 & e_1 & e_2 & e_3 & e_4 & e_5 & e_6 & e_7 \\
e_1 & e_1 & -e_0 & e_3 & -e_2 & e_5 & -e_4 & -e_7 & e_6 \\
e_2 & e_2 & -e_3 & -e_0 & e_1 & e_6 & e_7 & -e_4 & -e_5 \\
e_3 & e_3 & e_2 & -e_1 & -e_0 & e_7 & -e_6 & e_5 & -e_4 \\
e_4 & e_4 & -e_5 & -e_6 & -e_7 & -e_0 & e_1 & e_2 & e_3 \\
e_5 & e_5 & e_4 & -e_7 & e_6 & -e_1 & -e_0 & -e_3 & e_2 \\
e_6 & e_6 & e_7 & e_4 & -e_5 & -e_2 & e_3 & -e_0 & -e_1 \\
e_7 & e_7 & -e_6 & e_5 & e_4 & -e_3 & -e_2 & e_1 & -e_0 \\
\end{array}
\]

Table 6.1 Multiplication Table of Unit Octonions

There are many possible definitions for octonion multiplication. One of them which is called Fano Plane, usually used for unit octonions. A convenient diagram for remembering the product of unit octonions is given by Figure 6.2 which represent the multiplication table 6.1. There are seven points and seven lines in this diagram. The seven points corresponds to the seven standard basis elements of \( \mathbb{O} \), \( \{e_1, e_2, e_3, e_4, e_5, e_6, e_7\} \). Each line contains three points, and each of these three triples is equipped with a cyclic order [26] as indicated by the arrows. The rule is that if \( e_i, e_j, e_k \) are cyclically ordered in this way, they satisfy

\[
e_i^2 = e_j^2 = e_k^2 = -1,
\]

\[
e_i e_j = e_k = -e_j e_i.
\]

Thus, they give a copy of the quaternions inside the octonions.
Multiplication of two octonions can be written in a matrix form. Let \( u, v \in O \), then

\[
\begin{align*}
\mathbf{u} \star \mathbf{v} &= (u_0, u_1, u_2, u_3, u_4, u_5, u_6, u_7) \star (v_0, v_1, v_2, v_3, v_4, v_5, v_6, v_7) \\
&= \left[ u_0v_0 - u_1v_1 - u_2v_2 - u_3v_3 - u_4v_4 - u_5v_5 - u_6v_6 - u_7v_7 \\
u_1v_0 + u_0v_1 + u_2v_4 - u_4v_2 + u_5v_6 - u_6v_5 + u_3v_7 - u_7v_3 \\
u_2v_0 + u_0v_2 + u_3v_5 - u_5v_3 + u_6v_7 - u_7v_6 + u_4v_1 - u_1v_4 \\
u_3v_0 + u_0v_3 + u_4v_6 - u_6v_4 + u_7v_1 - u_1v_7 + u_5v_2 - u_2v_5 \\
u_4v_0 + u_0v_4 + u_1v_2 - u_2v_1 + u_5v_7 - u_7v_5 + u_6v_3 - u_3v_6 \\
u_5v_0 + u_0v_5 + u_2v_3 - u_3v_2 + u_6v_1 - u_1v_6 + u_7v_4 - u_4v_7 \\
u_6v_0 + u_0v_6 + u_3v_4 - u_4v_3 + u_7v_2 - u_2v_7 + u_1v_5 - u_5v_1 \\
u_7v_0 + u_0v_7 + u_4v_5 - u_5v_4 + u_1v_3 - u_3v_1 + u_2v_6 - u_6v_2
\right].
\end{align*}
\]

It is important to note that octonion multiplication is noncommutative as quaternions, but differently from quaternions, the octonions form a non associative multiplication algebra [5]. This can be verified by comparing (for example) products of octonions containing only single nonzero elements, such as

\[
\begin{align*}
(u_1 \star v_2) \star w_3 &= (uv) \star w_3 \\
&= -(uvw) \\
u_1 \star (v_2 \star w_3) &= u_1 \star (vw) \\
&= +(uvw).
\end{align*}
\]

In this example, we use the obvious notations \( u_1 = (0, u, 0, 0, 0, 0, 0, 0) \), \( (uv)_2 = (0, 0, uv, 0, 0, 0, 0, 0) \) (and so on) for octonions containing only a single nonzero element.
7 Conclusion

This thesis has attempted to provide a broad overview of quaternions and their applications. Particularly, application of Euler angles and quaternions in representing rotations have been explained and compared in this thesis. Euler angle sequences and quaternions are both common methods to use for rotation in $\mathbb{R}^3$. However, Euler angle sequences encounter the gimbal lock problem, but quaternions do not. That’s why quaternions are advantageous over Euler angles. Also we found out that quaternions are more efficient to create smooth interpolated motion by using SLERP.

This thesis have concluded and explored that beyond quaternions, there is another algebraic system, “octonions”. This number system is a generalization of quaternions. It is important to point out that when we upgraded from one dimension to another, we generally lost one specific property. For example, real numbers have an ordering relation, but when we generalize them to the complex numbers, these numbers can not be ordered. Similarly, complex numbers are commutative under multiplication but quaternions are not. And finally, quaternion multiplication has an associativity, but octonion multiplication is not associative. As a conclusion, quaternions are said to be much more efficient way in many respects compared to any rotation matrix systems.

References


