The use of quaternions in the computation of 3D rotations

Shayan Nezami

February 24, 2025

Contents

1 Introduction

1.1 What are quaternions?

Quaternions are a 4-dimensional number system that extend the complex numbers, in a similar way that the complex numbers extend the real numbers. Denoted $\mathbb{H}$, they are of the form:

$$a + b\mathbf{i} + c\mathbf{j} + d\mathbf{k}, \quad a, b, c, d \in \mathbb{R}. \tag{1}$$

In the modern day, they are used extensively in representing the rotations of 3-dimensional objects in computer systems, due to the relative computational ease of calculating such rotations, which only involves the multiplication of real numbers, and is not susceptible to problems like gimbal lock, unlike Euler angles, which are perhaps easier to visualise, and more widely used outside computational settings. [1]

1.2 The structure of this essay

After this brief introduction, some widely used key terms relating to quaternions will be defined, which will be essential to understanding the following chapters. Next, the additive and (more importantly) multiplicative properties of quaternions will be covered, which in effect govern how they are able to represent 3D rotations.

Having covered these areas, the mechanism in which they can be used for rotations will be discussed in depth, including relevant proofs and the properties which make them desirable to use. The essay will finish with comparisons to other methods of representing 3D rotations, as well as an example application of quaternions in a computational setting.

1.3 Definitions

1.3.1 Pure quaternions

A pure quaternion is a quaternion where there is no real component, meaning for a quaternion $a + b\mathbf{i} + c\mathbf{j} + d\mathbf{k}$, $a = 0$. [2]

They are also known as vector quaternions, as pure quaternions can be used to represent vectors in quaternion form, where $\mathbf{i}, \mathbf{j}, \mathbf{k}$ represent the x, y, z components of a vector.

For example, the vector $(2, 3, 1)$ can be represented as the quaternion $2\mathbf{i} + 3\mathbf{j} + \mathbf{k}$.

1.3.2 Unit quaternions

A unit quaternion is a quaternion with norm 1, i.e: $|q| = 1$ for a quaternion, q. [3]

1.3.3 The norm of a quaternion

The norm, $|q|$, of a quaternion, q, is defined as:

$$\begin{gathered} |q| = \sqrt{a^2 + b^2 + c^2 + d^2}, \\ \text{where } q = a + b\mathbf{i} + c\mathbf{j} + d\mathbf{k}. \end{gathered} \tag{2}$$

This could be thought of as similar to the magnitude of a vector, giving the theoretical "length" of the quaternion. [2]

This is referred to as the norm, as it is what each component of a quaternion needs to be divided by to normalise it (to get its corresponding unit quaternion - one with the same theoretical "direction").

For example, to normalise the quaternion

$$q = 2 + 3\mathbf{i} - 4\mathbf{j} + 5\mathbf{k}, \tag{3}$$

one would first find its norm:

$$|q| = \sqrt{2^2 + 3^2 + (-4)^2 + 5^2} = 3\sqrt{6}, \tag{4}$$

and then divide each of its components by the norm:

$$q_u = \frac{2}{3\sqrt{6}} + \frac{3}{3\sqrt{6}}\mathbf{i} + \frac{4}{3\sqrt{6}}\mathbf{j} + \frac{5}{3\sqrt{6}}\mathbf{k}. \tag{5}$$

It can be proven that this is indeed a unit quaternion, as

$$|q_u| = \sqrt{\left(\frac{2}{3\sqrt{6}}\right)^2 + \left(\frac{3}{3\sqrt{6}}\right)^2 + \left(\frac{4}{3\sqrt{6}}\right)^2 + \left(\frac{5}{3\sqrt{6}}\right)^2} = 1. \tag{6}$$

1.3.4 The conjugate of a quaternion

Every quaternion, q, has a conjugate, $\bar{q}$, which is defined as:

$$\begin{gathered} \bar{q} = a - b\mathbf{i} - c\mathbf{j} - d\mathbf{k}, \\ \text{where } q = a + b\mathbf{i} + c\mathbf{j} + d\mathbf{k}. \end{gathered} \tag{7}$$

Note that $\bar{q}$ can also be written q^*. [3]

2 Defining Properties of Quaternions

The algebraic field of quaternions is defined by certain core properties, which are explored below.

The additive properties are mostly included for context and completeness, while the multiplicative properties are essential for the rotation calculations that will follow - it is in effect these multiplicative properties that mean quaternions are able to compute 3D rotation.

Throughout this section it is to be assumed that $q_n \in \mathbb{H}$, and $a_n, b_n, c_n, d_n \in \mathbb{R}$.

2.1 Additive properties

2.1.1 Closure

Quaternions are considered closed upon addition. This means that:

$$q_1, q_2 \in \mathbb{H} \implies q_1 + q_2 \in \mathbb{H}. \tag{8}$$

In words, the addition of any two quaternions will necessarily produce a quaternion. [4]

2.1.2 Commutativity and associativity

Quaternion addition is both commutative,

$$q_1 + q_2 = q_2 + q_1, \tag{9}$$

and associative [4],

$$(q_1 + q_2) + q_3 = q_1 + (q_2 + q_3). \tag{10}$$

While this may seem obvious, this is not to be assumed for all number systems. In fact, the multiplication of quaternions, though associative, is not commutative. This will be further discussed later.

2.1.3 Calculating addition and subtraction

Summing and finding the difference of two quaternions is relatively straightforward - it simply consists of adding or subtracting each component of the quaternions: [4]

$$\begin{aligned} q_1 + q_2 &= (a_1 + b_1\mathbf{i} + c_1\mathbf{k} + d_1\mathbf{k}) + (a_2 + b_2\mathbf{i} + c_2\mathbf{k} + d_2\mathbf{k}) \\ &= (a_1 + a_2) + (b_1 + b_2)\mathbf{i} + (c_1 + c_2)\mathbf{k} + (d_1 + d_2)\mathbf{k}, \end{aligned} \tag{11}$$

$$\begin{aligned} q_1 - q_2 &= (a_1 + b_1\mathbf{i} + c_1\mathbf{k} + d_1\mathbf{k}) - (a_2 + b_2\mathbf{i} + c_2\mathbf{k} + d_2\mathbf{k}) \\ &= (a_1 - a_2) + (b_1 - b_2)\mathbf{i} + (c_1 - c_2)\mathbf{k} + (d_1 - d_2)\mathbf{k}. \end{aligned} \tag{12}$$

2.2 Multiplicative properties

2.2.1 Closure

As with addition, quaternions are considered closed upon multiplication, meaning:

$$q_1, q_2 \in \mathbb{H} \implies q_1 q_2 \in \mathbb{H}. \tag{13}$$

Thus, the product of any two quaternions will necessarily be a quaternion. [4]

2.2.2 Commutativity and associativity

Although quaternion multiplication, like addition, is associative: [4]

$$(q_1 \times q_2) \times q_3 = q_1 \times (q_2 \times q_3), \tag{14}$$

it is crucially not commutative: [2]

$$q_1 \times q_2 \neq q_2 \times q_1. \tag{15}$$

This may seem unintuitive at first, but it explains how multiple different rotations can be encoded within one quaternion even though the order of rotations matters (performing the same rotations in a different order produces a different result). [5] This will be explored in much more depth later.

Note that while the multiplication of two quaternions is not commutative, the multiplication of a quaternion and a real number is:

$$nq_1q_2 = q_1nq_2 = q_1q_2n \neq q_2q_1n, \quad n \in \mathbb{R}. \tag{16}$$

2.2.3 Distributivity

As with both real and complex numbers, quaternion multiplication is considered to be distributive over addition. This means that multiplying a quaternion by the sum of two other quaternions is equal to taking the sum of multiplying the quaternion with each quaternion being added:

$$q_1(q_2 + q_3) = q_1q_2 + q_1q_3. \tag{17}$$

2.2.4 Calculating multiplication

The first step in working out the product of two quaternions is distributing each component:

$$\begin{aligned}
&(a_1 + b_1\mathbf{i} + c_1\mathbf{j} + d_1\mathbf{k})(a_2 + b_2\mathbf{i} + c_2\mathbf{j} + d_2\mathbf{k}) \\
&= a_1a_2 + a_1b_2\mathbf{i} + a_1c_2\mathbf{j} + a_1d_2\mathbf{k} \\
&\quad + b_1a_2\mathbf{i} + b_1b_2\mathbf{i}^2 + b_1c_2\mathbf{ij} + b_1d_2\mathbf{ik} \\
&\quad + c_1a_2\mathbf{j} + c_1b_2\mathbf{ji} + c_1c_2\mathbf{j}^2 + c_1d_2\mathbf{jk} \\
&\quad + d_1a_2\mathbf{k} + d_1b_2\mathbf{ki} + d_1c_2\mathbf{kj} + d_1d_2\mathbf{k}^2.
\end{aligned} \tag{18}$$

Do note that the order of the quaternion components $(\mathbf{i}, \mathbf{j}, \mathbf{k})$ must stay the same, while the order of the real components is not relevant, as discussed in Section 2.2.2.

The result of this distribution, however, is not the final product - it can be further simplified using the defining quaternion multiplication rules, which will be explored in the following subsection.

2.3 The $\mathbf{i}, \mathbf{j}, \mathbf{k}$ components

2.3.1 The defining rules

While the following rules [5] could simply be considered to be multiplicative rules, they almost single-handedly define quaternions as a number system, and so are much more than simply multiplicative rules.

They were famously carved into a bridge by William Rowan Hamilton in 1843, after discovering that four, not three, dimensional numbers (quaternions) were in fact needed to describe 3D rotations: [1]

$$\mathbf{i}^2 = \mathbf{j}^2 = \mathbf{k}^2 = \mathbf{ijk} = -1, \tag{19a}$$

$$\mathbf{ij} = -\mathbf{ji} = \mathbf{k}, \tag{19b}$$

$$\mathbf{jk} = -\mathbf{kj} = \mathbf{i}, \tag{19c}$$

$$\mathbf{ki} = -\mathbf{ik} = \mathbf{j}. \tag{19d}$$

2.3.2 Why four dimensions are needed

By considering why four-dimensional numbers are necessary to encode three-dimensional rotations, the reasons behind each of the above rules will become clear.

It is trivial to consider why complex (two-dimensional) numbers cannot be used for three-dimensional rotations: they simply do not have the scope to even represent a three-dimensional point, let alone transform it. From this point, the natural solution seems to be the use of three-dimensional numbers.

Consider a three-dimensional number, t, where:

$$t = a + b\mathbf{i} + c\mathbf{j}, \quad a, b, c \in \mathbb{R}. \tag{20}$$

It would be possible to represent three-dimensional points in a number like t simply by assigning each of a, b, c to the point's x, y, z co-ordinates. For example, $(2, 3, 4)$ could be represented as $2 + 3\mathbf{i} + 4\mathbf{j}$.

However, where this number system fails is where its multiplicative rules are defined. It is logical to state that $\mathbf{i}^2 = \mathbf{j}^2 = -1$, but the problem comes with defining $\mathbf{ij}$.

Since $\pm\mathbf{i}$, $\pm\mathbf{j}$, and ± 1 all have a magnitude of 1 (a "distance" of 1 from the origin), the magnitude of $\mathbf{ij}$ must also be 1. This leaves three possibilities for the value of $\mathbf{ij}$: $\pm 1, \pm\mathbf{i}, \pm\mathbf{j}$.

Having established that:

$$\mathbf{i}^2 = \mathbf{j}^2 = -1, \tag{21}$$

exploring each case individually shows that none are valid:

$$\mathbf{ij} = \pm\mathbf{i} \implies \mathbf{j} = \pm 1, \tag{22a}$$

$$\mathbf{ij} = \pm\mathbf{j} \implies \mathbf{i} = \pm 1, \tag{22b}$$

$$\mathbf{ij} = \pm\mathbf{1} \implies \mathbf{i} = \mp\mathbf{j} \quad \text{or} \quad \mathbf{j} = \mp\mathbf{i}. \tag{22c}$$

This is because 1, $\mathbf{i}$ and $\mathbf{j}$ must be orthogonal, and in each of the three cases one component ends up not being orthogonal to another. For example, if $\mathbf{i} = -\mathbf{j}$, it follows that $\mathbf{i}$ and $\mathbf{j}$ are scalar multiples, and as such no longer orthogonal.

The only solution that can be found to this problem is equating the result of $\mathbf{ij}$ to a new component, $\mathbf{k}$, that is orthogonal to the real, $\mathbf{i}$, and $\mathbf{j}$ "axes". From this, we can derive the following rules:

$$\mathbf{i}^2 = \mathbf{j}^2 = \mathbf{k}^2 = \mathbf{ijk} = -1, \tag{23a}$$

$$\mathbf{ij} = -\mathbf{ji} = \mathbf{k}, \tag{23b}$$

$$\mathbf{jk} = -\mathbf{kj} = \mathbf{i}, \tag{23c}$$

$$\mathbf{ki} = -\mathbf{ik} = \mathbf{j}. \tag{23d}$$

The product of any two components $(\mathbf{i}, \mathbf{j}, \mathbf{k})$ yields the other, and reversing the order flips the sign of the result (e.g. $\mathbf{ij} = \mathbf{k} \implies \mathbf{ji} = -\mathbf{k}$), as a result of the non-commutativity of quaternion multiplication (Section 2.2.2).

Note that these equations (23) are the same as those defined for quaternions (19) on page 8. We have now in effect derived quaternions by establishing that a three-dimensional number system cannot be defined.

3 Calculating Rotations

3.1 Forming an equation

To calculate a rotation with quaternions, a multiplication of form $qr\bar{q}$ [3] needs to be carried out, where q is the unit quaternion defining the rotation, r is the vector being rotated (as a pure quaternion), and $\bar{q}$ is the conjugate quaternion of q. This multiplication will yield a pure quaternion, like r, that describes the rotated vector.

This may seem arbitrary at the moment, but it can be vaguely thought of as q describing a rotation on r, while $\bar{q}$ exists as a "balancing force". The true mechanism behind how this works, and how q can be determined to perform the desired rotation will be revealed later in this section.

3.1.1 Proof of the need for unit quaternions

For this rotation to work, q (and by definition also $\bar{q}$), need to be unit quaternions, so that the multiplication only rotates the vector quaternion r, and does not otherwise transform it.

Consider multiplying two quaternions and a vector quaternion, q_1, r, and q_2 (as needs to be done for rotations) such that

$$r' = q_1 r q_2. \tag{24}$$

The only way to ensure that $|r'| = |r|$ (which is needed to ensure the magnitude of the rotated vector is the same as that of the original one, meaning pure rotation has occurred, without any enlargements or stretches) is if $|q_1| = |q_2| = 1$, as multiplying quaternions also multiplies their norms ($|q_1 q_2| = |q_1||q_2|$). [2]

This can be proven as follows:

$$\begin{aligned} \text{let } q_1 &= a_1 + b_1\mathbf{i} + c_1\mathbf{j} + d_1\mathbf{k}, \\ q_2 &= a_2 + b_2\mathbf{i} + c_2\mathbf{j} + d_2\mathbf{k}. \end{aligned} \tag{25}$$

$$\implies q_1 q_2 = (a_1 + b_1\mathbf{i} + c_1\mathbf{j} + d_1\mathbf{k})(a_2 + b_2\mathbf{i} + c_2\mathbf{j} + d_2\mathbf{k}) \tag{26}$$

$$\begin{aligned} &= a_1a_2 + a_1b_2\mathbf{i} + a_1c_2\mathbf{j} + a_1d_2\mathbf{k} \\ &\quad + b_1a_2\mathbf{i} - b_1b_2 + b_1c_2\mathbf{k} - b_1d_2\mathbf{j} \\ &\quad + c_1a_2\mathbf{j} - c_1b_2\mathbf{k} - c_1c_2 + c_1d_2\mathbf{i} \\ &\quad + d_1a_2\mathbf{k} + d_1b_2\mathbf{j} - d_1c_2\mathbf{i} - d_1d_2 \end{aligned} \tag{27}$$

$$\begin{aligned} &= (a_1a_2 - b_1b_2 - c_1c_2 - d_1d_2) \\ &\quad + (a_1b_2 + a_2b_1 + c_1d_2 - c_2d_1)\mathbf{i} \end{aligned} \tag{28}$$

$$
\begin{aligned}
&+ (a_1c_2 - b_1d_2 + a_2c_1 + b_2d_1)\mathbf{j}\\
&+ (a_1d_2 + b_1c_2 - b_2c_1 + a_2d_1)\mathbf{k}\\
&\text{simplified using the rules in (19).}
\end{aligned}
$$

$$
\begin{aligned}
\implies |q_1q_2|^2 &= (a_1a_2 - b_1b_2 - c_1c_2 - d_1d_2)^2 &(29)\\
&\quad + (a_1b_2 + a_2b_1 + c_1d_2 - c_2d_1)^2\\
&\quad + (a_1c_2 - b_1d_2 + a_2c_1 + b_2d_1)^2\\
&\quad + (a_1d_2 + b_1c_2 - b_2c_1 + a_2d_1)^2\\
&= {a_1}^2{a_2}^2 + {a_1}^2{b_2}^2 + {a_1}^2{c_2}^2 + {a_1}^2{d_2}^2 &(30)\\
&\quad + {b_1}^2{a_2}^2 + {b_1}^2{b_2}^2 + {b_1}^2{c_2}^2 + {b_1}^2{d_2}^2\\
&\quad + {c_1}^2{a_2}^2 + {c_1}^2{b_2}^2 + {c_1}^2{c_2}^2 + {c_1}^2{d_2}^2\\
&\quad + {d_1}^2{a_2}^2 + {d_1}^2{b_2}^2 + {d_1}^2{c_2}^2 + {d_1}^2{d_2}^2\\
&\quad \text{after expansion and simplification.} &(31)
\end{aligned}
$$

$$
\begin{aligned}
|q_1|^2 &= {a_1}^2 + {b_1}^2 + {c_1}^2 + {d_1}^2, &(32)\\
|q_2|^2 &= {a_2}^2 + {b_2}^2 + {c_2}^2 + {d_2}^2.
\end{aligned}
$$

$$
\begin{aligned}
\implies (|q_1||q_2|)^2 &= |q_1|^2|q_2|^2\\
&= {a_1}^2{a_2}^2 + {a_1}^2{b_2}^2 + {a_1}^2{c_2}^2 + {a_1}^2{d_2}^2 &(33)\\
&\quad + {b_1}^2{a_2}^2 + {b_1}^2{b_2}^2 + {b_1}^2{c_2}^2 + {b_1}^2{d_2}^2\\
&\quad + {c_1}^2{a_2}^2 + {c_1}^2{b_2}^2 + {c_1}^2{c_2}^2 + {c_1}^2{d_2}^2\\
&\quad + {d_1}^2{a_2}^2 + {d_1}^2{b_2}^2 + {d_1}^2{c_2}^2 + {d_1}^2{d_2}^2.
\end{aligned}
$$

$$\therefore \quad (30) = (33) \implies |q_1q_2|^2 = (|q_1||q_2|)^2 \implies |q_1q_2| = |q_1||q_2|.$$

This shows that if unit quaternions were not used for q and $\bar{q}$ in this multiplication, the norm (magnitude) of r would change, which is not desired, as this represents an enlargement, but if unit quaternions were used (i.e. $|q| = |\bar{q}| = 1$) then $|qr\bar{q}| = |q||r||\bar{q}| = 1 \times |r| \times 1 = |r|$ as desired.

It also shows that $|r|$ does not have to be 1 (i.e. r does not have to be a unit quaternion), as the crucial detail is that the magnitude (norm) of r is preserved, which is the case given that $|q| = 1$ as shown above.

3.2 Finding the values of $\mathbf{qr\bar{q}}$

3.3 A formula for the result

A formula for the result of the rotation can be derived as follows:

```bash
#!/bin/bash

r=$1; xyz=(${r//,/ })

x_1=${xyz[0]}; y_1=${xyz[1]}; z_1=${xyz[2]}

x_2=0; y_2=0; z_2=0

shift
while [ "$#" -gt 0 ]; do
    angle=$(echo "3.141592654*$1/360" | bc -l)
    shift; axis=$1; shift

    a=$(echo "c($angle)" | bc -l)

    b=0; c=0; d=0

    case "$axis" in
        'x')
            b=$(echo "s($angle)" | bc -l) ;;
        'y')
            c=$(echo "s($angle)" | bc -l) ;;
        'z')
            d=$(echo "s($angle)" | bc -l) ;;
        *)
            break ;;
    esac

    x_2=$(echo "($a*$a + $b*$b - $c*$c - $d*$d)*$x_1 + 2*($b*$c - $a*$d)*$y_1 + 2*($a*$c + $b*$d)*$z_1" | bc -l)
    y_2=$(echo "($a*$a - $b*$b + $c*$c - $d*$d)*$y_1 + 2*($c*$d - $a*$b)*$z_1 + 2*($a*$d + $b*$c)*$x_1" | bc -l)
    z_2=$(echo "($a*$a - $b*$b - $c*$c + $d*$d)*$z_1 + 2*($a*$b + $c*$d)*$y_1 + 2*($b*$d - $a*$c)*$x_1" | bc -l)

    x_1=$x_2; y_1=$y_2; z_1=$z_2
done

echo $(printf %.5f $x_1) $(printf %.5f $y_1) $(printf %.5f $z_1)
```

References

[1] Wikipedia contributers. *Quaternion*. 30/01/2025. URL: `https://en.wikipedia.org/wiki/Quaternion` (visited on 21/02/2025).

[2] Justin Wyss-Gallifent. "MATH431: Quaternions". In: *University of Maryland* (December, 2021). URL: `https://math.umd.edu/~immortal/MATH431/book/ch_quaternions.pdf`.

[3] D. Rose. "Rotation Quaternions, and How to Use Them". In: *danceswithcode.net* (May, 2015). URL: `https://danceswithcode.net/engineeringnotes/quaternions/quaternions.html`.

[4] Angela Bennett and Volodymyr Kindratenko. *Introduction to Quaternionial Algebra*. 29/08/2000. URL: `https://users.ncsa.illinois.edu/kindr/emtc/quaternions/` (visited on 22/02/2025).

[5] Ben Eater and Grant Sanderson. *Visualizing quaternions*. While perhaps used less as a source, this helped immensely with gaining an intuitive understanding for quaternions. October, 2018. URL: `https://eater.net/quaternions` (visited on 17/02/2025).