The use of quaternions in the computation of 3D rotations

Shayan Nezami

April 2, 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:

$$a + b\mathbf{i} + c\mathbf{j} + d\mathbf{k}, \quad a, b, c, d \in \mathbb{R}, \quad \mathbf{i}^2 = \mathbf{j}^2 = \mathbf{k}^2 = -1. \tag{1}$$

The first term is referred to as the real part of the quaternion, while the other three components are imaginary. Each of these components (which can be thought of as axes in a 4-dimensional space) are orthogonal (perpendicular), and as such no real transformation on any component can map it onto another.

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 addition and multiplication of real numbers, and so is much more computationally efficient than other methods of calculating rotation, such as 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, and here the definition of - and need for - the quaternion number system will be derived.

Having covered these areas, the mechanism in which they can be used for rotations will be discussed in depth, including relevant proofs and how to construct rotations with quaternions. The essay will conclude by discussing the applications and desirable properties of quaternions.

The main body of the essay, where most of the explanations are, after the key definitions and properties are stated, starts from Section 2.3, on page 7, although the sections on conjugate quaternions and the commutativity of quaternion multiplication are particularly important (Sections 1.3.4 and 2.2.2 respectively).

Wherever any information, or inspiration, is taken from a source, it is cited in square brackets, with the numbers corresponding to the bibliography entries. All explanations and diagrams that are not followed by a citation are my original work.

This essay has been written with LaTeX using the VimTeX plugin. Figure 2 has been created with GeoGebra and Asymptote, while all of the other Figures have been made with TikZ.

1.3 Definitions

1.3.1 Pure quaternions

A pure quaternion is a quaternion that has 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 the $\mathbf{i}, \mathbf{j}$, and $\mathbf{k}$ components of the quaternion represent the x, y, and z components of a vector respectively.

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: [3]

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

Using the multiplicative rules for quaternions, which will be explained in Section 2.3, it can be proven that for any unit quaternion q, $q\bar{q} = \bar{q}q = 1$:

$$\begin{aligned}
q\bar{q} &= (a + b\mathbf{i} + c\mathbf{j} + d\mathbf{k})(a - b\mathbf{i} - c\mathbf{j} - d\mathbf{k}) \\
&= a^2 - ab\mathbf{i} - ac\mathbf{j} - ad\mathbf{k} \\
&\quad + ab\mathbf{i} - b^2\mathbf{i}^2 - bc\mathbf{ij} - bd\mathbf{ik} \\
&\quad + ac\mathbf{j} - bc\mathbf{ji} - c^2\mathbf{j}^2 - cd\mathbf{jk} \\
&\quad + ad\mathbf{k} - bd\mathbf{ki} - cd\mathbf{kj} - d^2\mathbf{k}^2 \\
&= a^2 + b^2 + c^2 + d^2 \\
&\quad - bc\mathbf{ij} - bd\mathbf{ik} - bc\mathbf{ji} - cd\mathbf{jk} - bd\mathbf{ki} - cd\mathbf{kj} \\
&= a^2 + b^2 + c^2 + d^2 \\
&\quad - bc\mathbf{k} + bd\mathbf{j} + bc\mathbf{k} - cd\mathbf{i} - bd\mathbf{j} + cd\mathbf{i} \\
&= a^2 + b^2 + c^2 + d^2 = 1, \text{ since } q \text{ is a unit quaternion.}
\end{aligned} \tag{8}$$

It can be shown that $\bar{q}q = q\bar{q}$, since $\bar{q} = -q + 2a$ (where a is the real component of q), so $\bar{q}$ is a real (-1) multiple of q plus a real number ($2a$), meaning that the multiplication of these two quaternions can be considered commutative, producing the same result regardless of order.

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:

$$q_1, q_2 \in \mathbb{H} \implies q_1 + q_2 \in \mathbb{H}. \tag{9}$$

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,

$$q_1 + q_2 = q_2 + q_1, \tag{10}$$

and associative [4],

$$(q_1 + q_2) + q_3 = q_1 + (q_2 + q_3). \tag{11}$$

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]

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

$$\begin{aligned} q_1 - q_2 &= (a_1 + b_1\mathbf{i} + c_1\mathbf{k} + d_1\mathbf{k}) - (a_2 + b_2\mathbf{i} + c_2\mathbf{k} + d_2\mathbf{k}) \\ &= (a_1 - a_2) + (b_1 - b_2)\mathbf{i} + (c_1 - c_2)\mathbf{k} + (d_1 - d_2)\mathbf{k}. \end{aligned} \tag{13}$$

2.2 Multiplicative properties

2.2.1 Closure

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

$$q_1, q_2 \in \mathbb{H} \implies q_1 q_2 \in \mathbb{H}. \tag{14}$$

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]

$$(q_1 \times q_2) \times q_3 = q_1 \times (q_2 \times q_3), \tag{15}$$

it is crucially not commutative: [2]

$$q_1 \times q_2 \neq q_2 \times q_1. \tag{16}$$

This may seem unintuitive at first, but it explains how multiple different rotations can be encoded within one quaternion, even though the order of rotation matters, meaning that performing the same rotations in a different order produces a different result. If multiplying quaternions in any order produced the same result, there would be no way for the product of multiple quaternions to represent a given number of rotations, as the true result of the rotation would be differ based upon the order of rotation, but the quaternion product would remain identical.

For example, consider three quaternions, q_x, q_y, q_z, encoding rotations in the x, y, and z axes respectively. The exact mechanism by which they encode rotations will be explained later, but a resultant quaternion, q, combining all three rotations into one, can be calculated as follows:

$$q = q_x q_y q_z. \tag{17}$$

This is only possible because the order of multiplication affects the products. Otherwise, this expression would be equal to, for example, $q_y q_x q_z$, which is a different rotation, as the result of a 3-dimensional rotation around multiple axes is dependant on the order of each individual rotation.

While the multiplication of two quaternions is not commutative, the multiplication of a quaternion and a real number is:

$$nq_1 q_2 = q_1 n q_2 = q_1 q_2 n \neq q_2 q_1 n, \quad n \in \mathbb{R}. \tag{18}$$

This is also true about the internal components of a quaternion: only multiplications between two imaginary quaternion components are non-commutative; multiplication between a real number and any imaginary quaternion component is commutative. Do also note that multiplications between the same component are always commutative by definition. For example, $\mathbf{i} \times \mathbf{i}$ is by definition commutative, as swapping the order of multiplication is meaningless.

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:

$$q_1(q_2 + q_3) = q_1q_2 + q_1q_3. \tag{19}$$

2.2.4 Calculating multiplication

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

$$\begin{aligned}
&(a_1 + b_1\mathbf{i} + c_1\mathbf{j} + d_1\mathbf{k})(a_2 + b_2\mathbf{i} + c_2\mathbf{j} + d_2\mathbf{k}) \\
&= a_1a_2 + a_1b_2\mathbf{i} + a_1c_2\mathbf{j} + a_1d_2\mathbf{k} \\
&\quad + b_1a_2\mathbf{i} + b_1b_2\mathbf{i}^2 + b_1c_2\mathbf{ij} + b_1d_2\mathbf{ik} \\
&\quad + c_1a_2\mathbf{j} + c_1b_2\mathbf{ji} + c_1c_2\mathbf{j}^2 + c_1d_2\mathbf{jk} \\
&\quad + d_1a_2\mathbf{k} + d_1b_2\mathbf{ki} + d_1c_2\mathbf{kj} + d_1d_2\mathbf{k}^2.
\end{aligned} \tag{20}$$

Do note that the order of the quaternion's imaginary 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 be considered to be simply multiplicative rules, they almost single-handedly define quaternions as a number system, and so deserve their own section.

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

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

$$\mathbf{ij} = -\mathbf{ji} = \mathbf{k}, \tag{21b}$$

$$\mathbf{jk} = -\mathbf{kj} = \mathbf{i}, \tag{21c}$$

$$\mathbf{ki} = -\mathbf{ik} = \mathbf{j}. \tag{21d}$$

2.3.2 Why four dimensions are needed

These rules could be explained by simply stating they are as they are by definition of the quaternion number system, but by considering why four-, rather than three-, dimensional numbers are needed, 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:

$$t = a + b\mathbf{i} + c\mathbf{j}, \quad a, b, c \in \mathbb{R}. \tag{22}$$

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 $\mathbf{i}^2 = \mathbf{j}^2 = -1$ (this ensures orthogonality to the real component), but the problem comes with defining $\mathbf{ij}$.

It is essential that each component is orthogonal to each other, as this is what allows each component to represent a perpendicular axis. Consider two components not being orthogonal to each other. This would imply that each value in one axis, or of one component, can be mapped to a value in the other by a scalar multiple. For example, if $\mathbf{i} = 2$, doubling any real value would yield its corresponding value in the $\mathbf{i}$ "axis". At this point, it would be illogical to refer to them as separate axes, as the $\mathbf{i}$ component would simply be equal to a multiplication by 2, and such placing it in the real axis, as shown in Figure 1.

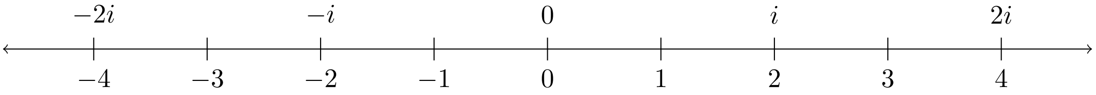

Figure 1: A number line visualising the mapping of the real and imaginary axes were they not orthogonal.

As such, the only way to have separate axes is for no component to be equal to a scalar multiple of another. $\mathbf{i}^2 = -1 \implies \mathbf{i} = \sqrt{-1}$ achieves this, as no real number can equal $\sqrt{-1}$, therefore ensuring the orthogonality of the two axes.

Since $\pm\mathbf{i}$, $\pm\mathbf{j}$, and ± 1 all have a magnitude of 1 (a "distance" of 1 from the origin when thinking of this graphically), the magnitude of $\mathbf{ij}$ must also be 1. This leaves three possibilities for the value of $\mathbf{ij}$: $\pm 1, \pm\mathbf{i}, \pm\mathbf{j}$, since these are the only numbers with a magnitude of 1 in this number system.

Having established that:

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

exploring each case individually shows that none are valid:

$$\begin{aligned}
\mathbf{ij} = \pm\mathbf{i} &\implies \mathbf{j} = \pm 1, && (24\text{a})\\
\mathbf{ij} = \pm\mathbf{j} &\implies \mathbf{i} = \pm 1, && (24\text{b})\\
\mathbf{ij} = \pm\mathbf{1} &\implies \mathbf{i} = \mp\mathbf{j} \quad \text{or} \quad \mathbf{j} = \mp\mathbf{i}. && (24\text{c})
\end{aligned}$$

This is because the real, $\mathbf{i}$ and $\mathbf{j}$ components must be orthogonal, and in each of the three cases above 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 that is orthogonal to the real, $\mathbf{i}$, and $\mathbf{j}$ "axes". In other words, for the $\mathbf{i}$ and $\mathbf{j}$ components to remain orthogonal, a new component is needed. We can call this new component $\mathbf{k}$, and define it as follows:

$$\mathbf{k}^2 = -1, \quad \mathbf{ij} = \mathbf{k}. \tag{25}$$

We now also need to define the multiplicative rules of $\mathbf{k}$ with $\mathbf{i}$ and $\mathbf{j}$. This is now fairly straightforward: we know that any component multiplied by another must form a separate component, and that the product of a component must be unique with any other component. Considering what we know about $\mathbf{i}$:

$$\begin{aligned}
\mathbf{i} \times 1 &= \mathbf{i}, && (26\text{a})\\
\mathbf{i} \times \mathbf{i} &= -1, && (26\text{b})\\
\mathbf{i} \times \mathbf{j} &= \mathbf{k}. && (26\text{c})
\end{aligned}$$

We can derive the equation for the multiplication of $\mathbf{k}$ and $\mathbf{i}$:

$$\mathbf{k} \times \mathbf{i} = \mathbf{j}. \tag{27}$$

This is because $\mathbf{i}$ multiplies with $1, \mathbf{i}$, and $\mathbf{j}$ to make $\mathbf{i}, -1$, and $\mathbf{k}$ respectively, so the only remaining component that $\mathbf{k}$ can multiply with $\mathbf{i}$ to make is $\mathbf{j}$. Do note that the order of multiplication here is reversed, and this is significant (in fact $\mathbf{ik} = -\mathbf{j}$) due to the non-commutativity of quaternion multiplication (Section 2.2.2), which is needed to enable them to encode rotations.

Repeating this process with $\mathbf{j}$ yields the following:

$$\mathbf{j} \times 1 = \mathbf{j}, \tag{28a}$$
$$\mathbf{j} \times \mathbf{j} = -1, \tag{28b}$$
$$\mathbf{j} \times \mathbf{i} = -\mathbf{k}, \tag{28c}$$
$$\mathbf{j} \times \mathbf{k} = \mathbf{i}. \tag{28d}$$

And so we have now defined all the multiplications of $\mathbf{k}$, while preserving orthogonality:

$$\mathbf{k} \times 1 = \mathbf{k}, \tag{29a}$$
$$\mathbf{k} \times \mathbf{k} = -1, \tag{29b}$$
$$\mathbf{k} \times \mathbf{i} = \mathbf{j}, \tag{29c}$$
$$\mathbf{k} \times \mathbf{j} = -\mathbf{i}. \tag{29d}$$

Putting these all together, we get:

$$\mathbf{i}^2 = \mathbf{j}^2 = \mathbf{k}^2 = \mathbf{ijk} = -1, \tag{30a}$$
$$\mathbf{ij} = -\mathbf{ji} = \mathbf{k}, \tag{30b}$$
$$\mathbf{jk} = -\mathbf{kj} = \mathbf{i}, \tag{30c}$$
$$\mathbf{ki} = -\mathbf{ik} = \mathbf{j}. \tag{30d}$$

The product of any two separate components $(\mathbf{i}, \mathbf{j}, \mathbf{k})$ yields the other, and reversing the order flips the sign of the result (e.g. $\mathbf{ij} = \mathbf{k} \implies \mathbf{ji} = -\mathbf{k}$).

Note that these equations (30) are the same as those defined for quaternions (21) on page 7. We have now in effect derived the quaternion number system by establishing that a three-dimensional number system cannot be defined.

2.3.3 Simplifying the multiplication

In Section 2.2.4 (page 7), the distributivity of quaternion multiplication was demonstrated through calculating the product of two sample quaternions, but with the multiplicative rules that we have defined

(21), we can now simplify this further:

$$\begin{aligned}
&(a_1 + b_1\mathbf{i} + c_1\mathbf{j} + d_1\mathbf{k})(a_2 + b_2\mathbf{i} + c_2\mathbf{j} + d_2\mathbf{k}) \\
&= a_1a_2 + a_1b_2\mathbf{i} + a_1c_2\mathbf{j} + a_1d_2\mathbf{k} \\
&\quad + b_1a_2\mathbf{i} + b_1b_2\mathbf{i}^2 + b_1c_2\mathbf{ij} + b_1d_2\mathbf{ik} \\
&\quad + c_1a_2\mathbf{j} + c_1b_2\mathbf{ji} + c_1c_2\mathbf{j}^2 + c_1d_2\mathbf{jk} \\
&\quad + d_1a_2\mathbf{k} + d_1b_2\mathbf{ki} + d_1c_2\mathbf{kj} + d_1d_2\mathbf{k}^2 \\
&= a_1a_2 + a_1b_2\mathbf{i} + a_1c_2\mathbf{j} + a_1d_2\mathbf{k} \\
&\quad + b_1a_2\mathbf{i} - b_1b_2 + b_1c_2\mathbf{k} - b_1d_2\mathbf{j} \\
&\quad + c_1a_2\mathbf{j} - c_1b_2\mathbf{k} - c_1c_2 + c_1d_2\mathbf{i} \\
&\quad + d_1a_2\mathbf{k} + d_1b_2\mathbf{j} - d_1c_2\mathbf{i} - d_1d_2 \\
&= (a_1a_2 - b_1b_2 - c_1c_2 - d_1d_2) \\
&\quad + (a_1b_2 + a_2b_1 + c_1d_2 - c_2d_1)\mathbf{i} \\
&\quad + (a_1c_2 - b_1d_2 + a_2c_1 + b_2d_1)\mathbf{j} \\
&\quad + (a_1d_2 + b_1c_2 - b_2c_1 + a_2d_1)\mathbf{k}.
\end{aligned} \tag{31}$$

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 (since the real component of r is 0, the product's real component will necessarily also be 0).

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" [5]. The true mechanism behind how this works, and how q can be determined to perform the desired rotation, will be explained through this chapter.

3.1.1 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:

$$r' = q_1rq_2. \tag{32}$$

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: $|q_1q_2| = |q_1||q_2|$.

This (that $|q_1q_2| = |q_1||q_2|$) can be proven as follows:

$$\begin{aligned}
\text{let } q_1 &= a_1 + b_1\mathbf{i} + c_1\mathbf{j} + d_1\mathbf{k}, \\
q_2 &= a_2 + b_2\mathbf{i} + c_2\mathbf{j} + d_2\mathbf{k}.
\end{aligned} \tag{33}$$

$$\begin{aligned}
\implies q_1q_2 = &(a_1a_2 - b_1b_2 - c_1c_2 - d_1d_2) \\
&+ (a_1b_2 + a_2b_1 + c_1d_2 - c_2d_1)\mathbf{i}
\end{aligned} \tag{34}$$

$$
\begin{aligned}
&+ (a_1c_2 - b_1d_2 + a_2c_1 + b_2d_1)\mathbf{j} \\
&+ (a_1d_2 + b_1c_2 - b_2c_1 + a_2d_1)\mathbf{k} \\
&\text{as shown in Section 2.3.3.}
\end{aligned}
$$

$$
\begin{aligned}
\implies |q_1q_2|^2 &= (a_1a_2 - b_1b_2 - c_1c_2 - d_1d_2)^2 && (35)\\
&\quad + (a_1b_2 + a_2b_1 + c_1d_2 - c_2d_1)^2 \\
&\quad + (a_1c_2 - b_1d_2 + a_2c_1 + b_2d_1)^2 \\
&\quad + (a_1d_2 + b_1c_2 - b_2c_1 + a_2d_1)^2 \\
&= {a_1}^2{a_2}^2 + {a_1}^2{b_2}^2 + {a_1}^2{c_2}^2 + {a_1}^2{d_2}^2 && (36)\\
&\quad + {b_1}^2{a_2}^2 + {b_1}^2{b_2}^2 + {b_1}^2{c_2}^2 + {b_1}^2{d_2}^2 \\
&\quad + {c_1}^2{a_2}^2 + {c_1}^2{b_2}^2 + {c_1}^2{c_2}^2 + {c_1}^2{d_2}^2 \\
&\quad + {d_1}^2{a_2}^2 + {d_1}^2{b_2}^2 + {d_1}^2{c_2}^2 + {d_1}^2{d_2}^2 \\
&\quad \text{after expansion and simplification.} && (37)
\end{aligned}
$$

$$
\begin{aligned}
|q_1|^2 &= {a_1}^2 + {b_1}^2 + {c_1}^2 + {d_1}^2, && (38)\\
|q_2|^2 &= {a_2}^2 + {b_2}^2 + {c_2}^2 + {d_2}^2.
\end{aligned}
$$

$$
\begin{aligned}
\therefore \quad (|q_1||q_2|)^2 &= |q_1|^2|q_2|^2 \\
&= {a_1}^2{a_2}^2 + {a_1}^2{b_2}^2 + {a_1}^2{c_2}^2 + {a_1}^2{d_2}^2 && (39)\\
&\quad + {b_1}^2{a_2}^2 + {b_1}^2{b_2}^2 + {b_1}^2{c_2}^2 + {b_1}^2{d_2}^2 \\
&\quad + {c_1}^2{a_2}^2 + {c_1}^2{b_2}^2 + {c_1}^2{c_2}^2 + {c_1}^2{d_2}^2 \\
&\quad + {d_1}^2{a_2}^2 + {d_1}^2{b_2}^2 + {d_1}^2{c_2}^2 + {d_1}^2{d_2}^2.
\end{aligned}
$$

$$(36) = (39) \implies |q_1q_2|^2 = (|q_1||q_2|)^2 \implies |q_1q_2| = |q_1||q_2|.$$

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 need to represent a unit vector), 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 quaternion of rotation

This section will explain how the values of the components in the formula $r' = qr\bar{q}$, where r' describes the co-ordinates of the point following rotation, can be found.

Since r is simply the position vector of the original point in the form of a pure quaternion (Section 1.3.1), and $\bar{q}$ is the conjugate quaternion of q, all that is left is to define q.

3.2.1 2D rotation with complex numbers

It will help to first consider performing 2-dimensional rotations [6] through complex number multiplication, as quaternion rotation can be thought of as just a "scaled-up" version of this.

Imagine a point (x_1, y_1), that is to be rotated by the angle θ in a 2-dimensional plane. Since we are just rotating this point, without scaling, we know that the distance of the rotated point, (x_2, y_2), from the origin will be the same as that of the original point. As such, we can consider both points on a circumference of a circle with radius r, where r is the distance of the points from the origin:

$$r = \sqrt{{x_1}^2 + {y_1}^2} = \sqrt{{x_2}^2 + {y_2}^2}. \tag{40}$$

The "aim" of this rotation is, given x_1, y_1, and θ, to be able to calculate x_2 and y_2.

We can start by defining the original and final points in complex number form, as $(x_1 + y_1\mathbf{i})$ and $(x_2 + y_2\mathbf{i})$ respectively. We can also consider two position vectors, u and v, from the origin to each of these points. Note that the x and y components of these vectors are the real and imaginary components of the complex numbers that represent them respectively. The final component to consider is the perpendicular line through u that intersects the point $(x_2 + y_2\mathbf{i})$. This is all illustrated in Figure 2.

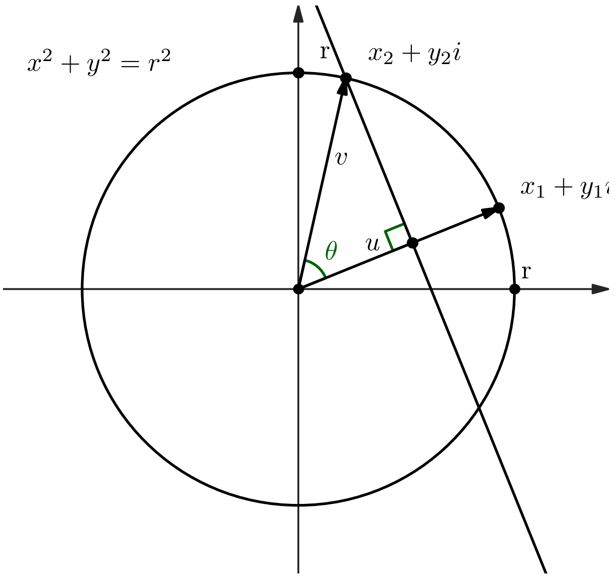

Figure 2: A visual representation of 2-dimensional rotation using complex numbers.

The position vector (and thereby the co-ordinates) of the rotated point, $(x_2 + y_2\mathbf{i})$, can therefore be found by adding the component of u up to the point it intersects the perpendicular line through it - this vector will be referred to as a - to the vector, b, from this point to the point $(x_2 + y_2\mathbf{i})$. Therefore, $(x_2 + y_2\mathbf{i}) = a + b$.

Defining θ as the angle between the vectors v and u:

$$\begin{aligned} \cos\theta = \frac{|a|}{r} &\implies |a| = r\cos\theta = |u|\cos\theta \\ &\implies a = u\cos\theta. \end{aligned} \tag{41}$$

To explain the above, $\cos\theta$ represents the ratio between the magnitude of a, and r, which is also the magnitude of u. Therefore, since a and u are parallel vectors (they are in the same direction), and $\cos\theta$ represents the ratio between them, $a = u\cos\theta$.

A very similar process can be followed to find b, using $\sin\theta$ instead of $\cos\theta$ as b is opposite rather than adjacent to θ:

$$\sin\theta = \frac{|b|}{r} \implies |b| = r\sin\theta = |u|\sin\theta. \tag{42}$$

However, while $|b| = |u|\sin\theta$, it does not follow here that $b = u\sin\theta$, as b is not parallel to u, but perpendicular by definition. Therefore, first we need to rotate u by $\frac{\pi}{2}$ radians to get it parallel with b. This can be done through multiplication with $\mathbf{i}$, as $\mathbf{i}$ is the perpendicular axis to the real numbers. In effect, this multiplication "swaps" the order of the x and y components, by moving the real numbers into the imaginary axis, and the imaginary components into the real axis:

$$\mathbf{i}^2 = -1 \implies (x + y\mathbf{i}) \times \mathbf{i} = (-y + x\mathbf{i}). \tag{43}$$

By multiplying u by $\mathbf{i}$ to make it parallel to b, it can then be multiplied by $\sin\theta$ to also make its magnitude equal to that of b (as shown above), so:

$$b = u \sin\theta \mathbf{i}. \tag{44}$$

Therefore, as $u = x_1 + y_1\mathbf{i}$:

$$\begin{aligned} v &= x_2 + y_2\mathbf{i} \\ &= a + b \\ &= u\cos\theta + u\sin\theta\mathbf{i} \\ &= (x_1 + y_1\mathbf{i})(\cos\theta + \sin\theta\mathbf{i}). \end{aligned} \tag{45}$$

This shows that a multiplication by $(\cos\theta + \sin\theta\mathbf{i})$ - a complex number with real component $\cos\theta$ and imaginary component $\sin\theta$ - rotates a point in 2-dimensional space by θ radians. It is clear that this is a pure rotation (i.e. the distance of the points from the origin remains unchanged), both by the diagram in Figure 2, and by the identity $\sin^2\theta + \cos^2\theta \equiv 1$:

$$\begin{aligned} &\text{let } c = \cos\theta + \sin\theta\mathbf{i} \\ \implies &|c| = \sqrt{\cos^2\theta + \sin^2\theta} = 1. \end{aligned} \tag{46}$$

Therefore, performing this multiplication cannot change the magnitude of the point, as the magnitude of the complex number performing the rotation is 1. Note that the magnitude of a complex number is defined similarly to that of a quaternion, and multiplying two complex numbers also multiplies their magnitudes, just like with quaternions (Section 3.1.1).

3.2.2 3D rotation with quaternions

Using quaternions to rotate 3-dimensional points is not so different to the process considered above, where complex numbers are used to rotate 2-dimensional points, and most of the principles considered there are also relevant here.

However, there are some key challenges. Firstly, it is not sufficient to simply rotate around the origin, like we did with 2-dimensional points - to rotate a 3-dimensional point, an axis of rotation also has to be defined. Plus, it is much more difficult to diagrammatically show this, since we cannot intuitively visualise 4-dimensional space.

We can attempt to visualise a unit quaternion in 4-dimensional space as in Figure 3, where the shaded 2-dimensional space represents the 3-dimensional imaginary $(\mathbf{i}, \mathbf{j}, \mathbf{k})$ components of the quaternions, while the vertical component of the sphere represents the real component.

In this representation, the $\mathbf{i}, \mathbf{j}, \mathbf{k}$ space is thought of as one component, perpendicular to the real numbers. Considering the $\mathbf{i}, \mathbf{j}, \mathbf{k}$ space as defining the axis we are rotating around (in the same way that pure quaternions represent vectors - Section 1.3.1), that axis will also be perpendicular to the real numbers, by definition of the orthogonality of the four quaternion components.

We can therefore think of a new diagram, Figure 4, similar to Figure 2, but with the axis perpendicular to the real numbers being the chosen axis to rotate around, instead of $\mathbf{i}$. This rotation axis, A, can be represented as a pure, unit quaternion, where:

$$A = x\mathbf{i} + y\mathbf{j} + z\mathbf{k}, \qquad \sqrt{x^2 + y^2 + z^2} = 1, \qquad x, y, z \in \mathbb{R}. \tag{47}$$

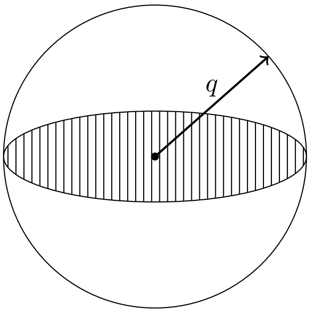

Figure 3: A 3-dimensional representation of quaternions. [7]

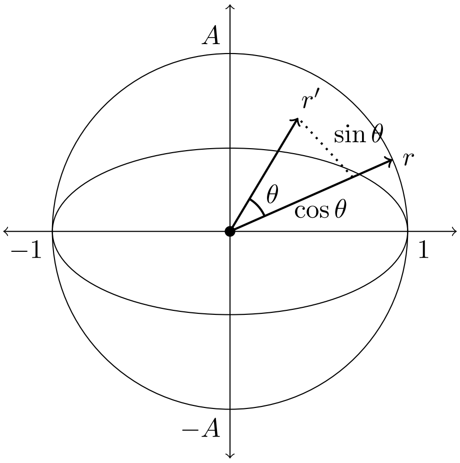

Figure 4: 4-dimensional rotations viewed on a unit sphere.

Figure 4 shows diagrammatically how the same method can be used for quaternion rotations as with 2-dimensional complex number rotations, but with a rotation around the axis A instead of around the origin. While in this diagram $|r| = 1$, as mentioned in Section 3.1, this is not necessary.

To rotate r to r', just like with complex numbers, r' can be considered to be the sum of two "vectors" (technically quaternions in this 4-dimensional space): the proportion of r in the direction of (parallel to) r, and the proportion of r perpendicular to r. As shown by the right-angled triangle formed in the diagram (and explained more thoroughly in Section 3.2.1 with complex numbers), the vector parallel to r is $r\cos\theta$, and the vector perpendicular to it is $Ar\sin\theta$. Here, A is used instead of $\mathbf{i}$, as it is being thought of as the perpendicular axis to the real numbers. Therefore, we can write r' (the rotated point) as:

$$\begin{aligned} r' &= r\cos\theta + Ar\sin\theta \\ &= (\cos\theta + A\sin\theta)r \\ &= (\cos\theta + (x\mathbf{i} + y\mathbf{j} + z\mathbf{k})\sin\theta)r \\ &= qr.^* \\ \implies q &= \cos\theta + (x\mathbf{i} + y\mathbf{j} + z\mathbf{k})\sin\theta,^* \end{aligned} \tag{48}$$

where q is the quaternion that r (a point being rotated in pure quaternion form) is multiplied by for a rotation of θ radians around the axis A.

Since A is a (pure) unit quaternion (which implies $x^2 + y^2 + z^2 = 1$), and $\sin^2\theta + \cos^2\theta \equiv 1$, this will

[*] This is not fully accurate, but should be considered so for the time being. The true form will be explained in Section 3.3.

guarantee that q will be a unit quaternion too, as desired:

$$\begin{aligned}|q| &= \sqrt{\cos^2\theta + (x\sin\theta)^2 + (y\sin\theta)^2 + (x\sin\theta)^2} \\ &= \sqrt{\cos^2\theta + \sin^2\theta(x^2+y^2+z^2)} \\ &= \sqrt{\cos^2\theta + \sin^2\theta} \\ &= 1.\end{aligned} \tag{49}$$

The use of sine for the imaginary axis of rotation and cosine for the real component can also be shown by looking at the sine and cosine functions themselves. For an angle of 0 (or any multiple of 2π - any full rotation), sine and cosine output 0 and 1 respectively. When the angle of rotation, θ, is 0, the desired effect is that qr yields r. If sine and cosine are used as given:

$$r' = qr^* = (\cos\theta + (x\mathbf{i} + y\mathbf{j} + z\mathbf{k})\sin\theta)r = 1r = r, \tag{50}$$

as desired. However, if the roles of sine and cosine were reversed, when $\theta = 0$:

$$r' = qr^* = (\sin\theta + (x\mathbf{i} + y\mathbf{j} + z\mathbf{k})\cos\theta)r = (x\mathbf{i} + y\mathbf{j} + z\mathbf{k})r. \tag{51}$$

This yields the product of r and axis of rotation, instead of just r as desired. Generalising, it is clear that (in the first quadrant) increasing θ increases $\sin\theta$ and decreases $\cos\theta$, and the "aim" when rotating is that increasing theta (in the first quadrant) should decrease the parallel (unchanged) component of r and increase the its perpendicular (rotated) component as a proportion of r'. This is also clear when looking at Figure 4. Therefore, it is logical that cosine should be used for the real component, and sine for the imaginary rotation axis.

However, there is a problem with the definition of a quaternion rotation above. In fact, $r' \neq qr, r' = qr\bar{q}$, as stated in Section 3.1. The following section will explain the reason for this, and how the formula above should be changed to reflect this.

3.3 The purpose of $\bar{q}$

While it is true, as shown above, that a multiplication by qr rotates r by θ radians around the axis, A, defined by q, it also performs a second, undesired, rotation, parallel to A. This is best demonstrated through a stereographic projection. [7] [8]

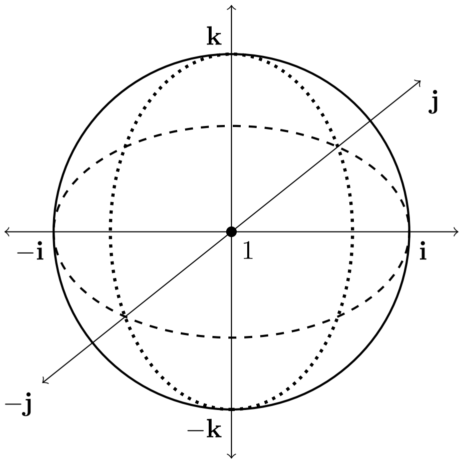

Figure 5: A stereographic projection of unit quaternions in 3-dimensional space. [7]

This stereographic projection (shown in Figure 5) maps every unit quaternion onto a point within 3-dimensional space. Note that the origin represents 1: the unit quaternion with real component 1 and therefore no imaginary component. The edge of this sphere that has been drawn represents every unit quaternion with a real component 0 (i.e. all the pure unit quaternions). The space inside the sphere represents quaternions with real component between 1 and 0, and the space outside the sphere represents the quaternions with negative real components. As a quaternion's real component tends towards -1, its representation in this diagram tends towards being infinitely displaced from the centre.

We are looking to rotate 3-dimensional points, that are represented as pure quaternions, and as such these points will all lie on the edge of the sphere, so we can place our focus on that part of the diagram. Although the quaternion being rotated does not need to be a unit quaternion, this diagram does not represent quaternions with any other magnitude, so the explanations will assume the point being rotated has magnitude 1. However, the explanations will still be relevant for any other pure quaternion, and the diagram can be modified to show quaternions of any magnitude by multiplying each component in the diagram by the desired magnitude. For example, for a magnitude of 2, on the diagram: $1 \to 2, -\mathbf{i} \to -2\mathbf{i}$, etc.

We can now imagine some multiplications of quaternions using this diagram. If we consider all the points on the sphere to have been left-multiplied by $\mathbf{i}$ (note the importance of the order of multiplication due to the non-commutativity of quaternion multiplication), using the multiplicative rules $\mathbf{ij} = \mathbf{k}$ and $\mathbf{ik} = -\mathbf{j}$ (Equation 21, Section 2.3), we can deduce that the point at $\mathbf{k}$ will be mapped to the point at $-\mathbf{j}$, the point at $-\mathbf{j}$ to $-\mathbf{k}$, the point at $-\mathbf{k}$ to $\mathbf{j}$, and finally the point at $\mathbf{j}$ to $\mathbf{k}$. On the diagram (Figure 5), this looks like an anticlockwise rotation on the dotted circle that connects the $\mathbf{k}$ and $\mathbf{j}$ axes.

Until this point, there have been no undesired effects - multiplying by $\mathbf{i}$ has resulted in a rotation perpendicular to, so around, the $\mathbf{i}$ axis, as desired. However, left-multiplying by $\mathbf{i}$ also causes a second transformation: one that is parallel to the $\mathbf{i}$ axis. More generally, a multiplication by any axis, A, causes a second rotation parallel to the axis. This can clearly be seen in the diagram, where left-multiplying by $\mathbf{i}$ also maps $\mathbf{i}$ to -1, -1 to $-\mathbf{i}$, $-\mathbf{i}$ to 1, and 1 to $\mathbf{i}$. While -1 cannot be explicitly seen in the diagram, it is the point of infinite distance from the centre of the unit circle, and so this rotation can be considered to "wrap-around" the diagram in a fourth dimension that we do not have access to. Figure 6 shows a visualisation of this. The arrows indicate the direction of motion upon left-multiplication by $\mathbf{i}$.

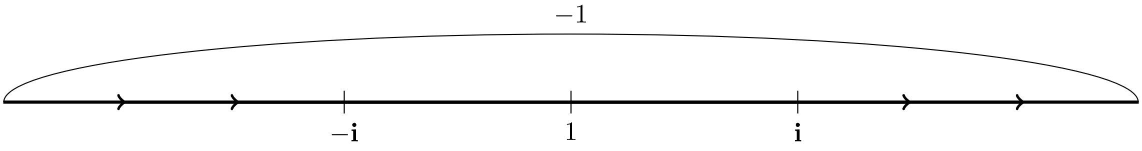

Figure 6: A representation of the theoretical position of -1 in Figure 5.

The same two rotations (both perpendicular and parallel to the axis) will occur with any axis of rotation, A, not necessarily just $\mathbf{i}$ or a single imaginary component. This behaviour is undesired, as the each multiplication should represent exactly one rotation around one axis. Therefore, the solution is to multiply the current expression, qr, by a third component, that will cancel out the rotation parallel to the axis, but not affect the rotation around it. Right-multiplying by $\bar{q}$ has this effect. Using the definition of $\bar{q}$ in Section 1.3.4:

$$\begin{aligned} q &= a + b_1\mathbf{i} + c_1\mathbf{j} + d_1\mathbf{k} \\ &= \cos\theta + (b_2\mathbf{i} + c_2\mathbf{j} + d_2\mathbf{k})\sin\theta \\ \implies \bar{q} &= a - b_1\mathbf{i} - c_1\mathbf{j} - d_1\mathbf{k} \\ &= \cos\theta - (b_2\mathbf{i} + c_2\mathbf{j} + d_2\mathbf{k})\sin\theta. \end{aligned} \tag{52}$$

To explain why right-multiplying by $\bar{q}$ has this effect, it is important to consider the implications of the non-commutativity of quaternion rotation: while the product of two different imaginary quaternion components differs based on the order of multiplication, this is not the case when a real number is involved in the multiplication (think of this as the real number "scaling" the imaginary component, rather than changing the axis on which it lies), or when one component is multiplied by itself (e.g. $\mathbf{i} \times \mathbf{i} = \mathbf{i} \times \mathbf{i}$).

If we consider qr as one point (as it is the pure quaternion formed following the first rotation of r), we can then consider the product of $\bar{q}$ with this pure quaternion qr. A left-multiplication by $\bar{q}$ would completely cancel out the effect of q, since $\bar{q}q = 1$ (Section 1.3.4) and quaternion multiplication is associative, i.e. $(\bar{q}q)r = \bar{q}(qr)$. Interpreting this through the stereographic visualisation of unit quaternions, left-multiplying qr by $\bar{q}$ would cause a rotation both parallel and perpendicular to the axis of rotation, opposite but equal in magnitude to the rotation a left-multiplication by q caused. More simply, a left-multiplication of qr by $\bar{q}$ yields r, cancelling out the effect of q.

The desired multiplication is one that does not affect the rotation perpendicular to the rotation axis, but causes a rotation with the opposite effect to the rotation parallel to it, in effect cancelling out the second, undesired, rotational effect of q. Right-multiplying by $\bar{q}$ has a similar effect to this. Since the non-commutativity of quaternion multiplication does not apply to multiplications of the same component, or to multiplications including a real number as one of the numbers being multiplied, this will have the same effect on the parallel axis as when left-multiplying by $\bar{q}$: one that cancels out the effect of q in this axis, as desired. This can be demonstrated with the previous example where the quaternions of multiplication, q, is $\mathbf{i}$. $q = \mathbf{i} \implies \bar{q} = -\mathbf{i}$, and right-multiplication by $\mathbf{i}$ maps $\mathbf{i}$ to -1, -1 to $-\mathbf{i}$, $-\mathbf{i}$, to 1, and 1 to $\mathbf{i}$, the exact opposite effect of left-multiplying by q, cancelling out this unwanted transformation.

However, the key here is how it affects the desired rotation, the rotation perpendicular to (around) the axis of rotation. In this case, the non-commutativity of multiplying different imaginary quaternion components in relevant, and so right-multiplying by $-\mathbf{i}$ causes a flip in the sign of the transformations created when compared to left-multiplying by $-\mathbf{i}$. This means that the theoretical circle between the $\mathbf{j}$ and $\mathbf{k}$ axes is rotated in the same direction as when it was left-multiplied by $\mathbf{i}$, mapping $\mathbf{k}$ to $-\mathbf{j}$, $-\mathbf{j}$ to $-\mathbf{k}$, $-\mathbf{k}$ to $\mathbf{j}$, and $\mathbf{j}$ to $\mathbf{k}$ on the stereographic projection. Again, this is the case with any axis, A, and so any rotation quaternion, q, but the transformation of points with the stereographic projection is simpler to visualise when a whole imaginary component is used as the axis of rotation.

While this right multiplication by $\bar{q}$, forming a final rotation expression of $r' = qr\bar{q}$, cancels out the undesired rotation q causes parallel to the axis of rotation, it does not leave the desired rotation perpendicular to (around) the axis untouched. It instead performs an identical rotation to q around the axis, in effect doubling the rotational effect of the expression. Therefore, defining q as $q = \cos\theta + (A)\sin\theta$ would actually rotate the point, r, 2θ, not θ, radians around the axis, A.

Therefore, we need to redefine q as:

$$q = \cos\frac{\theta}{2} + A\sin\frac{\theta}{2}, \tag{53}$$

where $qr\bar{q}$ defines the new position of r, a 3-dimensional point in pure quaternion form, when it is rotated θ radians around the axis, A, expressed as a pure unit quaternion.

3.4 The desirable properties of quaternions

While the field of quaternions was first defined by Hamilton, in 1843, it was not widely adopted as the standard for defining 3-dimensional geometry, and has only recently come into popularity, with the rise of computers, and the need for efficient computation. [8]

Euler angles are a key alternative to quaternions for representing 3-dimensional rotations. They work by defining the angle of rotation around each of the x, y, and z axes, and the order in which these rotations occur. They are typically much easier to visualise and intuitively understand than quaternions, however, are much less efficient and practical in a computational sense. The reasons for this become clear in this section, as the desirable properties of quaternions are explored.

3.4.1 The combination of multiple rotations

While there is a need to separately calculate three rotations around separate axes using Euler angles, one quaternion is sufficient to represent any possible 3-dimensional rotation. In fact, due to the non-commutativity of quaternion multiplication (Section 2.2.2), separate rotation quaternions can be multiplied together in their order of rotation to form one quaternion, representing the full rotation, by modifying the axis and angle of rotation accordingly.

This is particularly useful when converting from Euler angles to quaternions. This is a good way of increasing the ease-of-use of rotation systems, such as in a game engine, by allowing the programmer to define a rotation with Euler angles, but allowing the computer to calculate the rotation using quaternions.

For example, given Euler angles of $\frac{\pi}{4}$ radians in the y axis, $\frac{\pi}{2}$ in the x axis, and $-\frac{\pi}{4}$ in the z axis, where rotation is defined in the order y, x, z, the resultant quaternion of rotation, q_r, can be calculated as follows:

$$\begin{aligned}
& q = \cos\frac{\theta}{2} + (x\mathbf{i} + y\mathbf{j} + z\mathbf{k})\sin\frac{\theta}{2} \\
& q_r = q_y q_x q_z \\
\implies & q_y = \cos\frac{\pi}{8} + \sin\frac{\pi}{8}\mathbf{j} \\
& q_x = \cos\frac{\pi}{4} + \sin\frac{\pi}{4}\mathbf{i} \\
& q_z = \cos -\frac{\pi}{8} + \sin -\frac{\pi}{8}\mathbf{k} \\
\implies & q_r = (\cos\frac{\pi}{8} + \sin\frac{\pi}{8}\mathbf{j})(\cos\frac{\pi}{4} + \sin\frac{\pi}{4}\mathbf{i})(\cos -\frac{\pi}{8} + \sin -\frac{\pi}{8}\mathbf{k}) \\
& \quad = \frac{1}{2} + \frac{1}{2}\mathbf{i} + \frac{1}{2}\mathbf{j} - \frac{1}{2}\mathbf{k} \\
& \quad \text{after expansion and simplification.}
\end{aligned} \tag{54}$$

Do note here that while we can represent a rotation quaternion as $\cos\theta + (x\mathbf{i} + y\mathbf{j} + z\mathbf{k})\sin\theta$, this is only to allow a better mathematical understanding of the meaning behind the quaternion. In a purely computational sense, the quaternions are stored as four real numbers (one for the real component, and the other three as coefficients of the imaginary components), so trigonometric ratios do not need to be evaluated when calculating rotations with the quaternions produced.

This ability to combine individual rotations into one quaternion vastly reduces the number of calculations that need to be evaluated, and also reduces the possibility of numerical errors, which accumulate with each rotation, due to the computers not storing perfectly precise values for each component.

3.4.2 Animations

Another benefit of being able to use one quaternion for any rotation occurs when transitioning between two rotational states of an object. Quaternions make this simple, since the new rotation of the object can simply be calculated by using a fractional multiple of the same q (much more complicated algorithms for more realistic transitions do exist, but use the same principle).

For example, for an animation of a point with n frames, with r_n representing the position of the point in each frame, and r_0 its initial position, the frames would be calculated as:

$$\begin{aligned}
r_1 &= \frac{1}{n} q r_0 \bar{q} \\
r_2 &= \frac{2}{n} q r_0 \bar{q} \\
&\dots \\
r_n &= \frac{n}{n} q r_0 \bar{q} = q r_0 \bar{q}.
\end{aligned} \tag{55}$$

Note that to rotate a 3-dimensional object, such as a cube, relative to itself, the centre of the object should be considered the origin, and the rotations of each vertex calculated around that.

It is clear how this would be much more difficult with Euler angles, since rotations are defined in terms of three separate motions around separate axes. However, the main benefit of using quaternions here, is that no complex trigonometric ratios need to be evaluated once q has been found: evaluating $qr\bar{q}$ simply requires the addition and multiplication of real numbers, which is much less computationally complex. This is in contrast to Euler angles, where trigonometric ratios would need to be evaluated for every frame, as they are the basis upon which rotation is calculated, which would be very computationally demanding.

3.4.3 A full example

While expanding the expression $qr\bar{q}$ by hand can be very time-consuming, with many distributions and simplifications of imaginary quaternion components based upon their defined multiplicative rules, a computer does not need to calculate that for each rotation. After expanding and simplifying the general expression for $qr\bar{q}$ once, the result can be used as a formula for calculating rotations efficiently:

$$\begin{aligned} r' &= qr\bar{q} \\ &= (a + b\mathbf{i} + c\mathbf{j} + d\mathbf{k})(x\mathbf{i} + y\mathbf{j} + z\mathbf{k})(a - b\mathbf{i} - c\mathbf{j} - d\mathbf{k}) \\ &= ((a^2 + b^2 - c^2 - d^2)x + 2(bc - ad)y + 2(ac + bd)z)\mathbf{i} \\ &\quad + (2(ad + bc)x + (a^2 - b^2 + c^2 - d^2)y + 2(cd - ab)z)\mathbf{j} \\ &\quad + (2(bd - ac)x + 2(ab + cd)y + (a^2 - b^2 - c^2 + d^2)z)\mathbf{k} \\ &\quad \text{after much expansion and simplification.} \end{aligned} \tag{56}$$

Using the quaternion from Section 3.4.1, $q = \frac{1}{2} + \frac{1}{2}\mathbf{i} + \frac{1}{2}\mathbf{j} - \frac{1}{2}\mathbf{k}$, we can let $a = \frac{1}{2}, b = \frac{1}{2}, c = \frac{1}{2}, d = -\frac{1}{2}$. Assuming we are rotating the point $(1, 1, 1)$, we can also let $x = 1, y = 1, z = 1$. Substituting these values into the equation, all it takes is a few multiplications and additions to arrive at $r' = \mathbf{i} - \mathbf{j} - \mathbf{k}$. Therefore, the rotated point has co-ordinates $(1, -1, -1)$.

References

[1] Wikipedia contributors. *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.* September 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).

[6] Mathimaginatio. *Why Complex Multiplication Produces Rotation.* August 2022. URL: `https://www.youtube.com/watch?v=HozImwsr2ho` (visited on 23/03/2025).

[7] PenguinMaths. *How quaternions produce 3D rotation.* Inspiration for the diagram. July 2020. URL: `https://www.youtube.com/watch?v=jTgdKoQv738&t=565s` (visited on 25/03/2025).

[8] 3Blue1Brown. *Visualizing quaternions (4d numbers) with stereographic projection.* September 2018. URL: `https://www.youtube.com/watch?v=d4EgbgTm0Bg&t=1337s` (visited on 10/02/2025).