Project #1 : Quaternions

Nihar Patel UBIT: 50318506
Sahil Kapahi UBIT: 50317075
Raj Shah UBIT: 50321013
MAE 594: ROBOTICS-II

6th March, 2020

1 Introduction

Quaternions are absolutely fascinating and often under-appreciated number systems in math. Just as complex numbers are two dimensional extension of real numbers, Quaternions are 4-Dimensional extension of complex number. Their applications are mostly used for rotation in 3D dimension and quantum mechanics.

1.1 History

Quaternions were discovered on 16 October 1843 by an Irish Mathematician William Rowan Hamilton. He spent years trying to find a three dimensional number systems, but with no success, when he looked in 4 dimensions instead of 3 it worked. 3 Imaginary Dimensions describing space and the real numbers sitting perpendicular to them in 4-Dimension. He carved the crucial equation describing three imaginary units into the bridge which today there is plaque in his honor showing that equation.

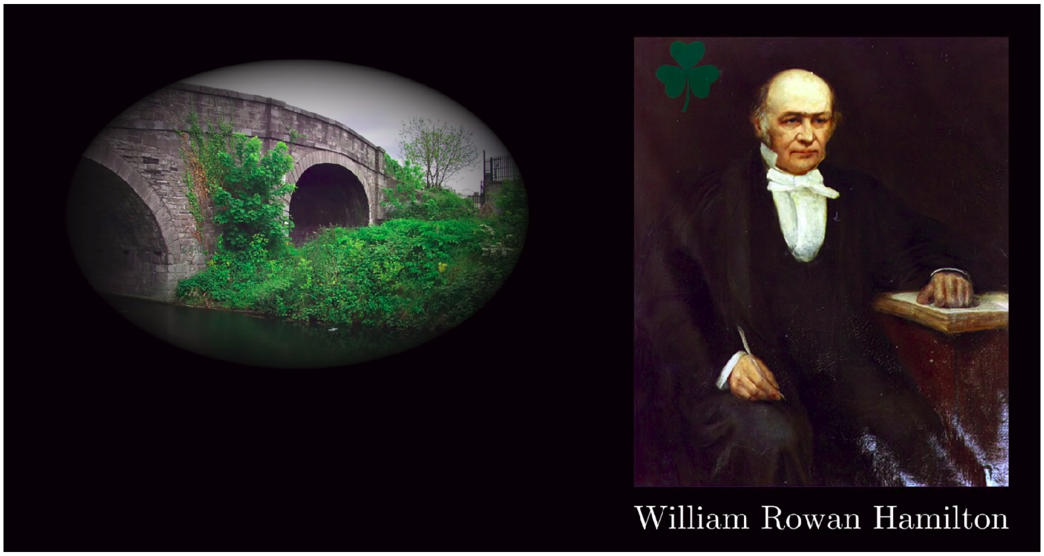

Figure 1: History - Motivation for Quaternion [1]

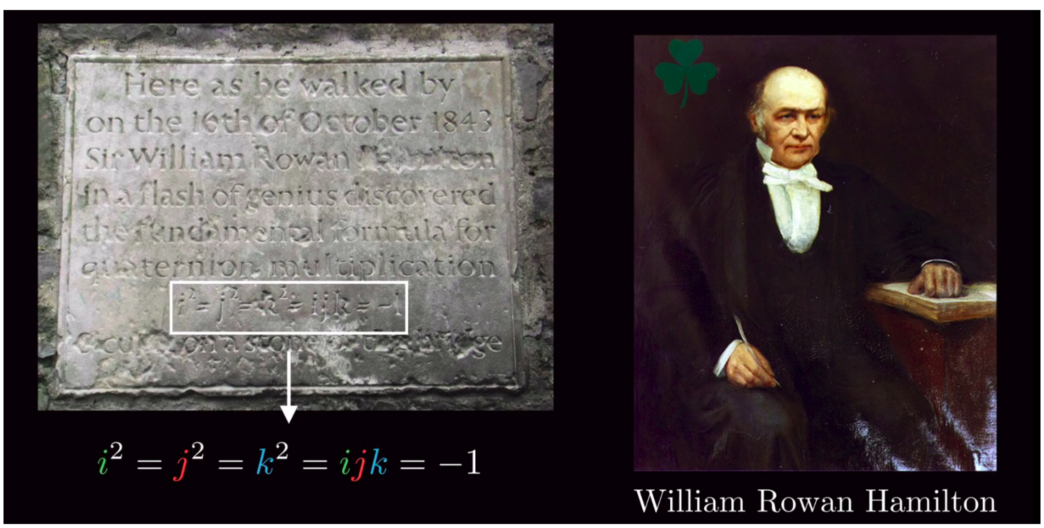

Figure 2: History - Plaque in honor of William Hamilton [1]

2 Background

Euler's equation can be used to represent a 2D point with a length and angle on a complex plane. Multiplication of 3 complex numbers implies a rotation in 2D space.

In order to extend to the rotation in the 3D space, William Hamilton initially studied it by adding an additional imaginary number, j to generalize complex numbers to 3D.

However, the set of 3 dimensional complex numbers is not closed under multiplication. For example, the multiplication of i and j cannot be represented as a form of a+ib+jc. If multiplication is closed, then there exist

$$a, b, c \in R$$

that satisfies ij=a+ib+jc
The equation $c^2 + 1 \neq 0$ gives the contradiction. There is no real number c to satisfy $(c)^2 + 1 = 0$

Later, William Hamilton realized that in order to perform multiplication of complex numbers it should be closed by adding an additional imaginary part,k and thus the number become 4-Dimensional. It is called as Quaternion

Following below is the equation of the Quaternion:

$$q = s + ix + jy + kz \tag{1}$$

2.1 Understanding

Quaternions have 4 dimensions (each quaternion consists of 4 scalar numbers), one real dimension and 3 imaginary dimensions. Each of these imaginary dimensions has a unit value of the square root of -1, but they are different square roots of -1 all mutually perpendicular to each other, known as i,j and k.

Quaternions form an interesting algebra where each object contains 4 scalar variables (sometimes known as Euler Parameters not to be confused with Euler angles), these objects can be added and multiplied as a single unit in a similar way to the usual algebra of numbers.

They are mostly used in practical applications like performing rotations(like graphics, robotics and orientation in 3D Space) and also in quantum mechanics.

They gave an elegant way to describe and compute 3D rotations which is computationally more efficient than other methods and also it gives less numerical errors than that of normal methods.

The main practical application of quaternion is to present 3D rotations.The majority of applications involve pure rotations, for this we restrict the quaternions to those with unit magnitude and we use only multiplications and not addition to represent a combination of different rotations.

2.2 Applications in Robotics Engineering:

In robotics industry, the quaternion representation often gives superior performance when compared to other representation. It provides computation advantage for robotics software like V-REP, MATLAB and Robotic Operating Systems (ROS)

When the forward kinematics and inverse kinematics are solved using quaternions it gives better computation efficiency for this application. Quaternions provide better geometrical than significance than the homogeneous representation.

For manipulator kinematics, quaternion method does not offer any advantage in computation as compared to the homogeneous method.

3 Arithmetic Operations of Quaternion

When we are using quaternions to represent rotations in 3 dimensions, then we restrict the quaternions to unit length and only use the multiplication operator.

Rotation in i, j, k and w = 1.

3.1 Addition of Quaternion Numbers

Since both scalars and vectors are present in a quaternion, the mathematical rules used to work with them are a combination of scalar and vector mathematics.

$$\begin{aligned} q_a &= a + i * b + j * c + k * d) \\ q_b &= e + i * f + j * g + k * h) \\ \boldsymbol{q_a} + \boldsymbol{q_b} &= (a+e) + (b+f) * i + (c+g) * j + (d+h)k \end{aligned} \tag{2}$$

3.2 Subtraction of Quaternion Numbers

Since both scalars and vectors are present in a quaternion, the mathematical rules used to work with them are a combination of scalar and vector mathematics.

$$\begin{aligned} q_a &= a + i * b + j * c + k * d) \\ q_b &= e + i * f + j * g + k * h) \\ \boldsymbol{q_a} - \boldsymbol{q_b} &= (a-e) + (b-f) * i + (c-g) * j + (d-h)k \end{aligned} \tag{3}$$

3.3 Multiplication of Quaternion Numbers

The multiplication rules for the imaginary operators are the same as for other numbers. We just put each quaternion in brackets and multiply out all the terms: (a + i b + j c + k d)*(e + i f + j g + k h).

When we are multiplying the imaginary operators we use the following rules:

- i*i = j*j = k*k = -1,
- i*j = k,
- j*i = -k,
- j*k = i,
- k*j = -i,
- k*i = j,
- i*k = -,

Since the multiplication of Quaternion is not commutative thus the order of the multiplication is significant. In other words $q_a * q_b \neq q_b * q_a$
The result is:

$\boldsymbol{q_a * q_b} = (a*e) - (b*f) - (c*g) - (d*h) + i(b*e + a*f + c*h - d*g) +$

j (a*g - b*h + c*e + d*f) + k .... (4)

3.4 Conjugate of Quaternion Numbers

Quaternion conjugate is defined by negating the vector part of the quaternion.

$$\begin{aligned} q_a &= a + i*b + j*c + k*d) \\ \boldsymbol{conj(q_a)} &= a - bi - cj - dk. \end{aligned} \tag{5}$$

The conjugate is useful because it has the following properties:

- $q_a' * q_b' = (q_b * q_a)'$. In this way we can change the order of the multiplications.
- $q^* q^{'} = a^2 + b^2 + c^2 + d^2 = real - number$. Multiplying a quaternion by its conjugate gives a real number. This makes the conjugate useful for finding the multiplicative inverse.
- For example, if we are using a quaternion q to represent a rotation then conj(q) represents the same rotation in the reverse direction.

3.5 3D Rotation with Quaternion

In 2D, the multiplication of two complex numbers implies 2D rotation. When $z = x + iy$ is multiplied by , the length of z' remains same ($|z| = |z'|$), but the angle of z' is added by θ

However, multiplying a quaternion p by a unit quaternion q does not conserve the length (norm) of the vector part of the quaternion p.
$p = i + j + k, \boldsymbol{V(p)} = i + j + k, |\boldsymbol{V(p)}| = \sqrt{3}$
q = i ($|q| = 1$)

pq = (i + j +k)i = -1 - k - j
$\boldsymbol{V(pq)}$ = j - k
$|V(p)| = \sqrt{1+1} = \sqrt{2}$
$\therefore |\boldsymbol{V(p)}| \neq |\boldsymbol{V(pq)}|$

Thus, we need a special multiplication for 3D rotations that is length-conserving transformation. For this reason, we multiply the unit quaternion q at the front of p and multiplying q^{-1} at the back of p, in order to cancel out the length changes. This special double multiplication is called "conjugation by q".
Thus, $qpq^{-1} = qpq^{*} =$
i (i + j +k)(- i) = (-1 + k -j)(- i)
= i - - k

4 Conclusion

Quaternions are an alternate way to describe orientation or rotations in 3D space using an ordered set of four numbers. They have the ability to uniquely describe any three-dimensional rotation about an arbitrary axis and do not suffer from gimbal lock. If you have an application where a sensor or object is capable of movement anywhere in 3D space, they are superior to Euler angles for tracking objects.

5 Animation

The animation of the quaternion is done in Robotic Operating System(ROS) using moveIt package. The moveIt package is used when you're dealing with controlling and operating of fix arm manipulator robot.

For our project, we have used the basic robot provided by the moveIt package. it is Panda robot from Franka Emika.

References

[1] "Visualizing quaternions (4d numbers) with stereo-graphic projection"

[2] Background of Quaternion. Source

[3] Arithmetic Properties of Quaternion. Source

[4] ROS - MoveIT Package Source