7 May 2024

Bachelor of Computer Science and Artificial Intelligence

Matrices and Linear Transformations

Quaternion Transformations and Their Applications

Group 5: India Antunez, Aya-Yasmine Belloum, Kye Electriciteh,

Trinidad Roca, & Makenna Warner

1. Abstract

This paper provides an in-depth analysis and exploration of the main aspects of quaternion transformations and its super cutting-edge approach for precise spatial orientation control in robotics.

This paper will commence with an elucidation of the critical role of inverse kinematics in attaining accurate and precise spatial manipulation. It then underscores the drawbacks of conventional techniques like Euler angles and stresses the deleterious impact of gimbal lock on robotic control. Then a transition is made so the paper can delve into the mathematical underpinnings and theoretical foundations of quaternions, explaining their representation in polar and rectangular forms and exploring their mathematical and theoretical roots. Key operations and properties of quaternions, including addition, multiplication, conjugate, magnitude, and inverse are investigated in detail, which is important to furnish the reader with a fundamental grasp of the use of quaternions in spatial orientation control.

Additionally, the study looks into the practical applications of quaternions in a range of fields, including as computer graphics, robotics, and physics simulations. By offering a solution to the common issue of gimbal lock in computer graphics, quaternions enhance animation realism and enable seamless rotation interpolation. Quaternions make difficult mathematical operations computationally efficient in physics simulations, enabling the high-fidelity replication of intricate physical phenomena. Quaternion transformations are bringing unprecedented accuracy and precision to robotic arm operations, changing the field of spatial orientation control in robotics. They do this by offering a robust and flexible framework.

The paper also looks at the special benefits that quaternions offer, like the capacity to encode extra information in quaternion representations and represent the same orientation with several quaternions. This adaptability makes calculations easier and data storage better, especially in contexts with limited resources. Researchers and practitioners can advance robotic spatial orientation control and create novel and cutting-edge applications in automation, manufacturing, healthcare, and other fields by utilizing the potential of quaternion transformations. All things considered, because they provide unmatched capability and performance in a variety of robotic applications, quaternion transformations constitute a paradigm leap in robotic control technique.

2. Introduction

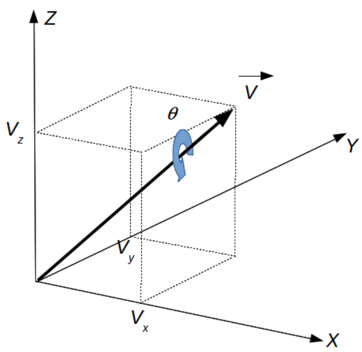
When trying to precise control over the spatial orientation and position of robotic arms, we must be directly linking to the discipline of inverse kinematics. Inverse kinematics takes charge in establishing the joint parameters that are required to put the robot's end-effector in the appropriate position, essentially involving the calculation of the angles and movements required at each joint; in order to position the robot's hand exactly where it needs to be. Contrary to forward kinematics, as they use known joint angles, this way deducing the end-effector's orientation and position, inverse kinematics operates in the opposite direction. Mainly aiming to identify the joint angles positioning the robot's hand at a specific area, which is critical for a wide range of applications, from simple automation chores to complex surgical robotics.

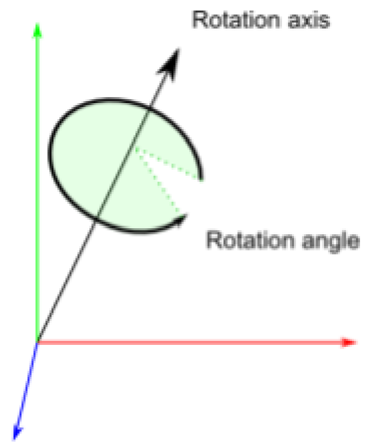
Traditionally, Euler angles have been used to represent these three-dimensional rotations, due to their straightforward method: dividing down rotations, creating three subsequent angular rotations along axes that make them an enticing alternative. However, we were wise to find their severe disadvantage, which limits the effectiveness of the Euler angles in robotic control. The main concern being the gimbal lock, this happens when two of the three rotation axes align, which could also result in the loss of one degree of freedom, as well as the ability to rotate around one axis. The gimbal lock certainly leaves us at a disadvantage by adding complexity and unpredictability to the solutions that could lead to problems when calculating the robotic movements and possibly failures in robotic activities.

In light of the current issues when calculating spatial orientation in robotics, this study will look at the idea of using quaternion transformations as an improved alternative for controlling position control. Furthermore, quaternions provide a complex and mathematically sound technique to express rotations, also avoiding the drawbacks inherent with Euler angles. Essentially, quaternions work by using a four-dimensional complex number system, this way they encode rotations in three dimensions compactly and efficiently, not only avoiding the problem of gimbal lock, but also allowing for smoother interpolations and simplifying the complex computations required in inverse kinematics. The transition to quaternions is more than just a technical preference; it is a deliberate enhancement to improve the reliability, accuracy, and efficiency of robotic manipulations across a wide range of applications.

3. Theory and Formulas

The key idea with Quaternions is that they are hypercomplex numbers that extend the concept of complex numbers into four dimensions. Quaternions consist of one real part and one imaginary part. These components make Quaternions particularly useful for representing rotations in 3 dimensional space. A rotation in 3D space can be described by an axis of rotation (featuring xyz components) and an angle of rotation (*see figures below), where the axis of rotation is represented by the imaginary part and the angle of rotation is related to the real component.





Quaternions can be expressed in Rectangular form as: $a + bi + cj + dk$, where a is the real part and bi, cj, and dk are the imaginary parts, with i, j, k being the fundamental Quaternion units.

Quaternions can be expressed in Polar form as: $q = cos(2\theta) + usin(2\theta)$, where θ is the angle of rotation and u is a unit vector representing the axis of rotation. U can be expressed as $xi + yj + zk$. Thus, for a unit vector axis of rotation $[x, y, z]$ and rotation angle θ, the Quaternion describing the rotation is: $cos(2\theta) + sin(2\theta)(xi + yj + zk)$.
Example: $cos(\Pi/4) + sin(\Pi/4)i$ represents a rotation over the x-axis for θ = $\Pi/2$.

Quaternions obey certain algebraic rules, which makes them good for representing rotations and orientations in three dimensional space. Here are some key operations and properties:

Addition and subtraction

Adding or subtracting quaternions is done component-wise, just like with vectors.

Example: $(w1 + xi1 + yj1 + zk1) + (w2 + xi2 + yj2 + zk2) =$

$$(w1 + w2) + (x1 + x2)i + (y1 + y2)j + (z1 + z2)k$$

Multiplication

$ijk = i^2 = j^2 = k^2 = -1$

$i \cdot j = ki \cdot j = k,\ j \cdot k = ij \cdot k = i,\ k \cdot i = jk \cdot i = j$

The product, called the Hamilton product, can be expressed as:

$$q1 \cdot q2 = (a1a2 - b1b2 - c1c2 - d1d2) + (a1b2 + b1a2 + c1d2 - d1c2)i$$
$$+ (a1c2 - b1d2 + c1a2 + d1b2)j + (a1d2 + b1c2 - c1b2 + d1a2)k$$

Note that Quaternion multiplication is not commutative, meaning $ab \neq ba$.

Multiplication is particularly useful for Quaternion rotation.

Conjugate

The conjugate of $q = w + xi + yj + zk$ is given by $\bar{q} = w - xi - yj - zk\bar{q} = w - xi - yj - zk$.

Conjunction is necessary for finding the inverse of a Quaternion and for normalising Quaternions.

Magnitude

$\|q\| = w2 + x2 + y2 + z2$

The magnitude is necessary for finding the inverse.

Inverse

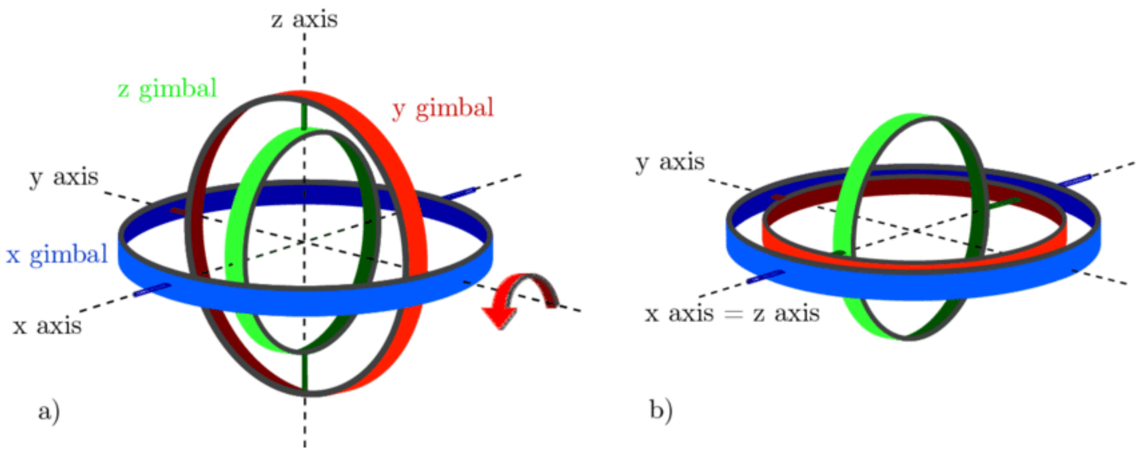
$q^{-1} = \bar{q}/\|q\|^2$

This is useful for finding the reciprocal of a Quaternion.

4. Application

The most common use case for quaternions is very simple: 3-dimensional angular transformation. More specifically, quaternions are really easily used by computers. The ability to accurately represent any 3D rotation with just a cheap linear transformation is very valuable to many fields using digital technology. Robotics, computer graphics, sensor technology, and physics simulations being a few among many of such valuable applications. The wonderful thing about quaternions—and they are indeed wonderful—is how much one can do with them. Each of these different applications use quaternions because of a specific instance or set of benefits that specifically lends to the nature of the work.

For example, almost every computer graphics system uses quaternions as a mainstay of rotation for one main reason: getting around gimbal lock. As mentioned in the introduction, gimbal lock is a mechanistic problem where two axes in a three dimensional gimbal align, preventing rotation on one axis altogether. Fig. B demonstrates how rotating around the global z axis is impossible from this configuration.



The quaternion solves this issue by including a fourth axis to rotate around, allowing for every possible transform to be properly reached by using a 3D hyperplane of the 4D space. This is especially important for computer graphics because time is an ever-present factor in most use cases. Using quaternions to overcome gimbal lock allows for linear interpolation between values directly, without having to do really complex calculations to rotate and un-rotate a traditional gimbal system multiple times.

Applications in scientific research and physics simulations rely on a different property of quaternions—arguably the most important one of all. Being just a set of complex numbers, quaternions can easily be represented as matrices. Either a 2x2 complex matrix, or a 4x4 real matrix. For example, the quaternion *a* + *b***i** + *c***j** + *d***k** can be written as both of the following matrices:

$$\begin{bmatrix} a+bi & c+di \\ -c+di & a-bi \end{bmatrix} \qquad \begin{bmatrix} a & -b & -c & -d \\ b & a & -d & c \\ c & d & a & -b \\ d & -c & b & a \end{bmatrix}$$

4x4 matrices are the most common representation, as they have been used extensively for many years and have very optimised methods for dealing with them, meaning mathematical operations are quite cheap for computers to perform. This is excellent for large scale simulations where thousands of operations need to be performed every frame. Traditionally, quaternions are used for CPU operations, meaning most uses are purely mechanical. Recent advances in GPU technology, however, are making quaternion operations even more performant, allowing for increasingly complex software that capitalises on the GPUs parallelisation capabilities (Gavilan, 2016).

Lastly, a quirky aspect of quaternions that all fields benefit from is the flexibility of moving from a high dimensional representation to a lower one. More specifically, multiple quaternions can represent the same orientation. This is useful because it simplifies calculations in the sense that only partial solutions are necessary to find a valid result. Furthermore, extra data can be encoded in numerical different quaternions that are dimensionally equivalent. For example, quaternions represent the same rotation as any equivalent scalar multiples, similarly to how vectors of different magnitude can have the same direction. So saying, a representation of a robotic joint by a quaternion could not only use the quaternion for rotation values, but also store something like torque drive in the same quaternion without adding or losing any data.

This is excellent for many applications where processing power is limited, as it is essentially free data storage.

5. Conclusion

Quaternions are a powerful and efficient tool for controlling spatial orientation– from computer animations to robotic arms to deep water GPS systems. Quaternions are an interesting culmination of what was covered in the *Matrices and Linear Transformations* course, as their use incorporates vectors, matrices, linear transformations (i.e. rotations), and solving linear equations. We have established that Quaternions, $q = a + bi + cj + dk,$ are a combination of vectors and matrices, with real, $a,$ and imaginary parts $bi + cj + dk$. The real part, $a,$ acts as a scalar, akin to the scalar components of vectors. The imaginary part $bi + cj + dk,$ represents a vector in three-dimensional space, similar to how a matrix represents a transformation in linear algebra. Additionally, quaternions can be used to solve linear equations: for instance, the original position is represented by a set of vectors, the desired position is represented by another set of vectors. We can use a linear equation to solve for the quaternion multiplication that maps the vectors onto each other. Quaternions are advantageous, as they can be expressed more concisely than matrices. They are also beneficial as they avoid the Gimbal Lock limitations of Euler angles.

In the future, quaternions will be helpful in many fields, including virtual reality, medical robots, autonomous vehicles, and aeronautical engineering, where precise orientation control is essential. In virtual reality, quaternions are used to accurately track the orientation of headgear and controllers, providing realistic experiences and total immersion. Because quaternions accurately reflect vehicle orientations, which are crucial for both safe and efficient operation, they are beneficial for autonomous cars. This facilitates accurate control and navigation. Medical robotics uses quaternions to precisely control imaging and surgical tools, enhancing the accuracy of procedures like minimally invasive surgeries and diagnostic imaging. In aerospace engineering, quaternions are utilized for spacecraft orientation control, where precise adjustments are needed for trajectory corrections and attitude control to ensure the success of space missions. Quaternions' adaptability and consistency make them valuable for accurate control and effective performance in spatial manipulation tasks as technology develops.

6. Bibliography

1. *Rotations, Orientation, and Quaternions - MATLAB & Simulink*. (n.d.). https://www.mathworks.com/help/nav/ug/rotations-orientation-and-quaternions.html. Accessed 15 April 2024.
2. *Two worked out examples of rotations using quaternions.* https://www.imsc.res.in/~knr/131129workshop/writeup_knr.pdf. Accessed 12 April 2024.
3. Gavilan, D. (2016, September 13). *Performance of quaternions in the GPU – Metail Tech*. https://tech.metail.com/performance-quaternions-gpu/. Accessed 15 April 2024.
4. Mukundan, R., & Mukundan, R. (2012). Quaternions. *Advanced Methods in Computer Graphics: With examples in OpenGL*, 77-112. Accessed 22 April 2024.
5. Voight, J. (2021). *Quaternion algebras* (p. 885). Springer Nature. Accessed 18 April 2024.

7. Annexes

[Link to the jupyter notebook]