Dual Quaternions Robotics: A) The 3R Planar Manipulator

Mahmoud Gouasmi*, Belkacem Gouasmi and Mohammed Ben-Ahmed-Dahou

Algeria Structural Mechanics Research Laboratory, Mechanical Engineering Department, Blida University, Algeria

Abstract: Kinematics analysis studies the relative motions, such as, first of all, the displacement in space of the end effector of a given robot, and thus its velocity and acceleration, associated with the links of the given robot that is usually designed so that it can position its end-effector with a three degree-of-freedom of translation and three degree-of-freedom of orientation within its workspace. This chapter presents mainly, on the light of both main concepts, the first being the screw motion or/ and dual quaternions kinematics while the second concerns the classical ‘Denavit and Hartenberg parameters method’ the direct kinematics of a planar manipulator.

First of all, examples of basic solid movements such as rotations, translations, their combinations and general screw motions are studied using both (4x4) matrices rigid body transformations and dual quaternions so that the reader could compare and note the similarity of the results obtained using one or the other method. Both dual quaternions technique as well as its counterpart the classical ‘Denavit and Hartenberg parameters method’ are finally applied to a three degree of freedom (RRR) planar manipulator. Finally, we and the reader, can observe that the two methods confirm exactly one another by giving us the same results for each of the examples and applications considered, while noting that the fastest, simplest more straightforward and easiest to apply method, is undoubtedly the one using dual quaternions. As a result this work may as well act as a beginners guide to the practicability of using dual-quaternions to represent the rotations and translations ie: or any rigid motion in character-based hierarchies.

We must emphasize the fact that the use of Matlab software and quaternions and / or dual quaternions in the processing of 3D rotations and/or screw movements is and will always be the most efficient, fast and accurate first choice. Dual quaternion direct kinematics method could be generalised, in the future, to more complicated spatial and/ or industrial robots as well as to articulated and multibody systems.

Keyword: Dual Quaternions, Forward Kinematics, Homogeneous Matrix, Screw Motion.

1. INTRODUCTION

Many research students have a great deal of trouble understanding essentially what quaternions are [1-3] and how they can represent rotation. So when the subject of dual-quaternions is presented, it is usually not welcomed with open arms. Dual-quaternions are a break from the norm (i.e., matrices) which we hope to entice the reader into supporting willingly to represent their rigid transforms. The reader should walk away from this chapter with a clear understanding of what dual-quaternions are and how they can be used [4].

First we begin with a short recent and related work that emphasises the power of dual-quaternions:

The dual-quaternion has been around since 1882 [5-7] but has gained less attention compared to quaternions alone; while the most recent work which has taken hold and has demonstrated the practicality of dual-quaternions, both in robotics and computer graphics can be resumed in: - Kavan [8] demonstrated the advantages of dual-quaternions in character skinning and blending. - Ivo [9] extended Kavan’s work with dual-quaternions and t-tangents as an alternative method for representing rigid transforms instead of matrices, and gives evidence that the results can be faster with accumulated transformations of joints if the inferences per vertex are large enough. - Selig [10] addressed the key problem in computer games. - Vasilakis [11] discussed skeleton-based rigid-skinning for character animation. - Kuang [12] presented a strategy for creating real-time animation of clothed body movement. -Pham [13] solved linked chain inverse kinematic (IK) problems using Jacobian matrix in the dual-quaternion space. -Malte [14] used a mean of multiple computational (MMC) model with dual-quaternions to model bodies. - Ge [15] demonstrated dual-quaternions to be an efficient and practical method for interpolating three-dimensional motions. -Yang -Hsing [16] calculated the relative orientation using dual-quaternions. - Perez [17] formulated dynamic constraints for articulated robotic systems using dual-quaternions. - Further reading on the subject of dual numbers and derivatives is presented by Gino [18].

In the last three decades, the field of robotics has widened its range of applications, due to recent developments in the major domains of robotics like kinematics, dynamics and control, which leads to the sudden growth of robotic applications in areas such as manufacturing, medical surgeries, defense, space vehicles, under-water explorations etc.

To use robotic manipulators in real-life applications, the first step is to obtain the accurate kinematic model [19]. In this context, a lot of research has been carried out in the literature, which leads to the evolution of new modeling schemes along with the refinement of existing methodologies describing the kinematics of robotic manipulators.

Screw theory based solution methods have been widely used in many robotic applications. The elements of screw theory can be traced to the work of Chasles and Poinsot [20, 21]. In the early 1800’s and Whittaker [22]. Using the theorems of Chasles and Poinsot as a
starting point, Robert S. Ball developed [23] a complete
type of screws which he published in 1900.
Throughout the development of kinematics, numerous
mathematical theories [24] and tools have been
introduced and applied. The first pioneer effort for
kinematic modeling of robotic manipulators was made
by Denavit and Hartenberg in introducing a consistent
and concise method to assign reference coordinate
frames to serial manipulators, allowing the (4×4)
homogeneous transformation matrices to be used (in
1955) [25], followed by Lie groups and Lie Algebra by
J.M. Selig and others, [26-28] and quaternions and
dual quaternions introduced by Yang and Freudenstein
(1964) [29], see also Bottema and Roth (1979) [30] and
McCarthy (1990) [31]. The original D–H parameter
method has many counterparts: Distal variant, proximal
variant, ...to name but a few. There even exist different
options for these counterparts.

In this method, four parameters, popularly known
as D–H parameters, are defined to provide the
generic description to serial mechanisms. Out of the
four, two are known as link parameters, which describe
the relative location of two attached axes in space.
These link parameters are: The link length (a) and the
link twist (α). (See appendix 11, 3.)

The remaining two parameters are described as
joint parameters, which describe the connection of any
link to its neighboring link. These are the joint offset (dl)
and the joint angle (θ).

Modeling the movement of the rigid body by the
type of the helicoidal axis: a combination of an
amount of rotation about and an amount of translation
along a certain axis, hence the term helicoidal axis is
used in various fields such as computer vision
and biomechanics. The application of this theory in the field of
robotics is taking more and more space. We can
consider the motion of a joint segment as a series of
finite displacements. In this case the movement is
characterized by an angle of rotation about an
amount of translation along an axis defined in space by
its position and its orientation. This axis is referred to as
the finite helicoidal axis (FHA), because of the
discretization of the movement into a series of
placements. On the other hand and by taking the
continuity of the movement into account, this
movement will be characterized by a rotational speed
(angular velocity) about and translation speed along an
axis defined by the instantaneous position and
orientation in space. One speaks in this case of an
instantaneous helicoidal axis (IHA). The application of
the helicoidal theory with its two versions (FHA and
IHA) is used to describe and understand the joint
movement, and to study in biomechanics, for example,
the different positioning techniques of protheses. Thus
there are several methods to estimate the helicoidal
axis from a set of points representing a rigid body. Any
displacement of a rigid body is a helicoidal motion
which may be decomposed into an angular rotational
movement about and a linear translational movement
along a certain axis in 3D space. The methods differ in
the way of mathematically representing these two
movements. These movements can be expressed
using rotation matrices and translation vectors,
homogeneous matrices [32-34] unit quaternions, dual
quaternions, ... The two representations; using (3x3)
matri ces or (4x4) homogeneous matrices and dual
quaternions will be simultaneously used for all and
each examples or applications studied so that
comparisons for each case could be done.

2. DUAL QUATERNIONS

2.1. «Product Type» Dual Quaternions

The dual quaternions have two forms thus two
readings which are complementary and simultaneous:
The first is the << product type >> description:

\[
T_{G} = \{ T_{R} + \varepsilon \frac{T_{X} \times T_{R}}{2} \} \quad \text{With:} \quad T_{R} = \left\{ \cos \frac{\psi}{2}, n \sin \frac{\psi}{2} \right\} = \left\{ \cos \frac{\psi}{2}, \sin \frac{\psi}{2} \cdot n_{x}, \sin \frac{\psi}{2} \cdot n_{y}, \sin \frac{\psi}{2} \cdot n_{z} \right\}
\]

\[
T_{F} = (0, T_{X}, T_{Y}, T_{Z}) = (0, \{T\})
\]

Then, the transformation is:

\[
T_{G} = \{ T_{R} + \varepsilon \frac{T_{X} \times T_{R}}{2} \} = \left\{ \cos \frac{\psi}{2}, n \sin \frac{\psi}{2} \right\} + \varepsilon \left\{ -\frac{T_{X}}{2} \sin \frac{\psi}{2}, \frac{T_{X}}{2} \cdot n_{y} \sin \frac{\psi}{2} + \frac{T_{X} \times T_{R}}{2} \right\} << \text{product type} >>
\] (1)

2.2. «Dual Type» Dual Quaternions

Indeed a general transformation, screw type, can be
also described using dual angles and dual vectors and
have therefore the following form << Dual type >>:

\[
\hat{T} = \left\{ \cos \frac{\theta}{2}, \sin \frac{\theta}{2} \cdot n \right\} = \left\{ \cos \frac{\theta}{2}, \sin \frac{\theta}{2} \cdot n \right\} + \varepsilon \left\{ -\frac{d}{2} \sin \frac{\theta}{2}, \left\{ m \sin \frac{\theta}{2} + \frac{\theta}{2} \cdot n \cos \frac{\theta}{2} \right\} \right\} << \text{dual type} >>
\] (2)

It is defined by the dual angle \( \hat{\theta} \) and the dual vector \( \hat{n} \), the rotation being represented by the angle \( \theta \)
around the axis \( n = (n_{x}, n_{y}, n_{z}) \) of norm 1, and a
translation \( d \) along the same vector \( n \).

The vector \( m = (m_{x}, m_{y}, m_{z}) \) is the moment of
rotation of \( n \) about the origin of reference \( (O, x, y, z) \); it is
named the moment of the axis \( n \), with: \( \hat{\theta} = \theta + \varepsilon \frac{d}{2} \) with \( d \) being the amplitude of the translation along the
dual vector \( \hat{n} = n + \varepsilon m \) with \( m = p \times n \) (the green
e vector see Figure 1) that defines the vector according to
Plücker coordinates, \( p \), (the blue vector), being the
vector that gives the position of \( n \), (the red vector),
using the vector \( \text{OO} \) (see Figure 1).

The parameters of the transformation, the angle \( \theta \),
the axis of rotation \( n \), the magnitude of the translation \( d \)
and the moment \( m \) are the four characteristics of all,
and every 3D rigid body transformation (4x4)
imatrix, a screw motion or a helicoidal movement of any
kind (or type).
Figure 1: Helicoidal or screw motion.

Note that this form resembles that used for classics quaternions; using the dual angle and the dual unitary vector instead of the classical ones.

And as a matter of fact: The screw displacement is the dual angle $\theta + \varepsilon d$, along the screw axis defined by the dual vector $\vec{l}$ or $\vec{d}$ or in our case $\vec{\omega} = \mathbf{n} + \varepsilon \mathbf{m}$; such that we will obtain (respecting the rules of derivation and multiplication of dual numbers), dual vectors, quaternions and dual quaternions (see appendix 10.2, and eq (24)):

$$\tilde{\vec{r}} = \left\{ \cos \frac{\theta}{2}, \sin \frac{\theta}{2} \right\} = \left[ \cos \frac{\theta}{2} - \varepsilon \frac{d}{2} \sin \frac{\theta}{2}, \left( \sin \frac{\theta}{2} + \varepsilon \left( \mathbf{n} + \varepsilon \mathbf{m} \right) \right) \right] = \cos \frac{\theta}{2} - \varepsilon \frac{d}{2} \sin \frac{\theta}{2}, \mathbf{n} \sin \frac{\theta}{2} + \varepsilon \left( \mathbf{m} \right) \left( \cos \frac{\theta}{2} \mathbf{n} \sin \frac{\theta}{2} + \varepsilon \left( \frac{d}{2} \sin \frac{\theta}{2} \sin \frac{\theta}{2} + \varepsilon \left( \frac{d}{2} \cos \frac{\theta}{2} \right) \right) \right)$$

The geometric interpretation of these quantities is related to the screw-type motion. The angle $\theta$ is the angle of rotation around $\mathbf{n}$, the vector unit $\mathbf{n}$ represents the direction of the rotation axis. The element $d$ is the translation or the displacement amplitude along the vector $\mathbf{n}$, the vector moment of the vector axis $\mathbf{n}$ relative to the origin of the axes. The vector $\mathbf{m}$ is an unambiguous description of the position of an axis in space, in accordance with the properties of Plücker coordinates defining lines in space.

This form gives another interesting use: Whereas the classics quaternions can only represent rotations whose axes go (or pass) through the origin $O$ of the coordinates system $(O, x, y, z)$, the dual quaternions can represent rotations about arbitrary axes in space, translations as well as any combination of both these two basic movements.

These two forms $<< product type >>$ eq (1) or $<< dual type >>$ eq (2) represent the same motion that describe the same movement ‘the screw motion’.

3. EXAMPLE 1: ROTATIONS REPRÉSENTED BY QUATERNIONS

Let’s apply two successive rotations to a rigid body: the first one of amplitude $\theta_1 = \frac{\pi}{2}$ around the axis $Ox$ followed by a second rotation of the same amplitude $\theta_2 = \frac{\pi}{2}$ around the $Oy$ axis. Using quaternions the first rotation will be written; since $\frac{\theta_1}{2} = \frac{\pi}{4}$ then $\cos \frac{\theta_1}{2} = \sin \frac{\theta_1}{2} = \frac{\sqrt{2}}{2} q_1 = (\frac{\sqrt{2}}{2}, \frac{\sqrt{2}}{2}, 0, 0)$; having $\frac{\theta_2}{2} = \frac{\pi}{4}$ then $\cos \frac{\theta_2}{2} = \sin \frac{\theta_2}{2} = \frac{\sqrt{2}}{2}$. The second rotation will have the form: $q_2 = (\frac{\sqrt{2}}{2}, 0, \frac{\sqrt{2}}{2}, 0)$ . The final composition of the two movements will be given by the quaternion $q$ such that: $q = q_2 . q_1 = (\frac{\sqrt{2}}{2}, 0, \frac{\sqrt{2}}{2}, 0)$ .

It is then easy to extract both the amplitude and the resulting axis of the rotation from the result $q$:

$$\cos \frac{\theta}{2} = \frac{1}{2} \text{ and } \sin \frac{\theta}{2} = \frac{\sqrt{3}}{2} ; \text{ wich implies the first solution } \theta = + 120^\circ, \text{ around the unitary axis } \mathbf{n} = \frac{1}{\sqrt{3}} \left(1\right)$$

or $\cos \frac{\theta}{2} = \frac{1}{2} \text{ and } \sin \frac{\theta}{2} = - \frac{\sqrt{3}}{2} ; \text{ wich implies a second solution } \theta = - 120^\circ, \text{ around the unit axis } (- \mathbf{n}) = \frac{1}{\sqrt{3}} \left(1\right)$

In fact the two solutions represent the same and similar solution since for any $q$ we have $q (\theta, n) = q (-\theta, -n)$

Using our classical (3x3) rigid transformations we get:

$$R_{21} = R_2 R_1 = \begin{pmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 \\ -1 & 0 & 0 \end{pmatrix} = \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & -1 \end{pmatrix} \begin{pmatrix} -1 & 0 & 0 \\ -1 & 0 & 0 \end{pmatrix}$$

Here it is very important to note that unlike the quaternion method we cannot extract the needed results easily and straightforwardly but we must follow a long and sometimes complicated process (determinant, trace, antisymmetry, angle and axis of rotations signs, axis/angle (or conversions to Olinde Rodrigues (Axe, Angle) parameters) …
Whichever used technique we will find: A rotation of 
\[ \theta = \frac{2\pi}{3} = 120^\circ \] around the unit axis \( n = \frac{1}{\sqrt{3}} \begin{pmatrix} 1 \\ 1 \\ -1 \end{pmatrix} \)

To show the anticommutativity of the product let’s do the inverse and start by the second rotation instead:
\[ q_1 = q_2 = \begin{pmatrix} \frac{\sqrt{3}}{2} \\ \frac{\sqrt{2}}{2} \\ 0 \end{pmatrix}, \quad q_2 = \begin{pmatrix} \frac{\sqrt{2}}{2} \\ \frac{\sqrt{2}}{2} \\ 0 \end{pmatrix} = \begin{pmatrix} 1 \\ \frac{\sqrt{3}}{2} \\ \frac{\sqrt{3}}{2} \end{pmatrix} = q_1 = \begin{pmatrix} \frac{\sqrt{3}}{2} \\ -\frac{\sqrt{3}}{2} \\ 0 \end{pmatrix} \]

and that will imply \( \theta = 120^\circ \) around the axis \( n = \frac{1}{\sqrt{3}} \begin{pmatrix} 1 \\ 1 \\ -1 \end{pmatrix} \),
or \( \theta = -120^\circ \) around the axis \( (-n) = \frac{1}{\sqrt{3}} \begin{pmatrix} -1 \\ -1 \\ 1 \end{pmatrix} \).

This of course will imply that: \( q_1 = q_2 \neq 2 \cdot q_1 \).

Using matrices:
\[
\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix}, \quad \begin{pmatrix} 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \\ -1 & 0 & 0 & 0 \end{pmatrix} \neq R = R_2 R_1
\]

wich implies:

A rotation of \( \theta = \frac{2\pi}{3} = 120^\circ \) around the unit axis \( n = \frac{1}{\sqrt{3}} \begin{pmatrix} 1 \\ 1 \\ -1 \end{pmatrix} \)
equiv to a rotation of \( \theta = -\frac{2\pi}{3} = -120^\circ \)
around the unit axis \( n = -\frac{1}{\sqrt{3}} \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix} \).

Using MATLAB (See Appendix 11, 1.) we can calculate easily both the two quaternions multiplications: \( q = q_1 q_2 \) and \( q = q_2 q_1 \) and the two equivalent products of matrices \( R_2 R_1 \) and \( R_1 R_2 \).

4. IMPORTANT NOTES: WHAT ABOUT TRANSLATIONS?

We must recall that rotations act on translations, the reverse being not true; in fact when multiplying by blocks: For a rotation followed by a translation:
\[
\begin{pmatrix} 1 & t \\ 0 & 1 \end{pmatrix} \begin{pmatrix} R \\ 0 \end{pmatrix} = \begin{pmatrix} R & t \end{pmatrix}; \quad \text{the rotation is not affected by the translation.}
\]

While for a translation followed by a rotation:
\[
\begin{pmatrix} R \\ 0 \end{pmatrix} \begin{pmatrix} 1 & t \\ 0 & 1 \end{pmatrix} = \begin{pmatrix} R & Rt \end{pmatrix}; \quad \text{the translation is affected by the rotation.}
\]

When translations are performed first we can thus assume that the translation vector of the resulting matrix product; \( R T \) act as the translation vector \( t \) of a rotation followed by a translation. Or more generally speaking considering two six degree of freedom general rigid body transformations \( T_1 \) followed by \( T_2 \) we will have:
\[
T_2 \cdot T_1 = \begin{pmatrix} R_2 \\ 0 \end{pmatrix} \begin{pmatrix} t_2 \\ 1 \end{pmatrix} = \begin{pmatrix} R_1 \\ 0 \end{pmatrix} \begin{pmatrix} t_1 \\ 1 \end{pmatrix} = \begin{pmatrix} R_2R_1 \\ 0 \end{pmatrix} \begin{pmatrix} R_2t_1 + t_2 \\ 1 \end{pmatrix} = \begin{pmatrix} R \\ 0 \end{pmatrix} \begin{pmatrix} t \\ 1 \end{pmatrix}
\]

The translation vector \( t \) of the product of the two transformations is \( \{ t_1 = R_2 t_1 + t_2 = \begin{pmatrix} R_2 \\ 0 \end{pmatrix} \begin{pmatrix} t_1 \\ 1 \end{pmatrix} \}
\]

The same analysis as the last one could then be done whatever the order and the number of the successive transformations being performed over the rigid body: The final result of the products of all the undertaken rigid body transformations will be finally the helicooidal, the helical or the screw motion given by the (4x4) matrix:
\[
[T] = T_n ... T_i ... T_2.T_1 = \begin{pmatrix} R \\ 0 \end{pmatrix} \begin{pmatrix} t \\ 1 \end{pmatrix}
\]

with \( T_i \) representing either a rotation, a translation, a rotation followed by a translation, a translation followed by a rotation or even simply a no movement (ie: the 4x4 identity matrix I).

5. SCREW MOTION

Any screw motion would be given by the following (4x4) matrix \( [T] \):
\[
\begin{pmatrix} I & u \\ 0 & 1 \end{pmatrix} \begin{pmatrix} R(\theta, n) \ & \ 0 \\ \frac{\theta p}{2\pi} n & \ 1 \end{pmatrix} \begin{pmatrix} I \\ 0 \end{pmatrix} = \begin{pmatrix} R(\theta, n) \ & \ 0 \\ \frac{\theta p}{2\pi} n + (I - R(\theta, n)u) & \ 1 \end{pmatrix} = [T]
\]

The middle matrix is a screw about a line through the origin; that is, a rotation around the axis \( n \) followed by a translation along \( n \). The outer matrices conjugate the screw and serve to place the line at an arbitrary position in space. The parameter \( p \) is the pitch of the screw; it gives the distance advanced along the axis for every complete turn, exactly like the pitch on the thread of an ordinary nut or bolt. When the pitch is zero the screw is a pure rotation, positive pitches correspond to right hand threads and negative pitches to left handed threads.

To show that a general rigid motion is a screw motion, we must show how to put a general transformation into the form derived above. The unit vector in the direction of the line \( n \) is easy since it must be the eigenvector of the rotation matrix corresponding to the unit eigenvalue. (This fails if \( R = I \), that is if the motion is a pure translation). The vector \( u \) is more difficult to find since it is the position vector of any point on the rotation axis. However we can uniquely specify \( u \) by requiring that it is normal to the rotation axis. So we impose the extra restriction that \( n.u = 0 \). So to put the general matrix \( [R(\theta, n) \begin{pmatrix} I \\ 0 \end{pmatrix} \] into the above form we must solve the following system of linear equations:
\[
\begin{pmatrix} \frac{\theta p}{2\pi} n + (I - R(\theta, n)u = t \quad \text{Now} \ n. R u = n.u = 0, \text{since the rotation is about} \ n. \quad \text{So we can dot the above equation with} \ n \text{to give:} \quad 0 = n.( t - \frac{\theta p}{2\pi} n) \text{this enables us to find the pitch:}
\]

\[ p = \frac{2\pi}{\theta} n. t \quad \text{All we need to do now is to solve the equation system:} \quad (I - R) u = (t - (n. t) n); \quad \text{This is}
\]
possible even though det \((I - R) = 0\), since the equations will be consistent.

This entire analysis established through this long paragraph concerning the helicoidal motion or rigid (4x4) transformation matrix \([T]\) is contained in only one line enclosed in its counterpart dual quaternion \(\hat{T}\) of the form:

\[
T = \begin{bmatrix}
\cos \frac{\theta}{2} & \sin \frac{\theta}{2} & \epsilon \\
\frac{\epsilon}{2} & \frac{\epsilon}{2} & -\frac{n}{2} \cos \frac{\theta}{2} \\
-\frac{n}{2} \sin \frac{\theta}{2} & \frac{n}{2} \sin \frac{\theta}{2} & \epsilon \\
\frac{\epsilon}{2} & \frac{\epsilon}{2} & -\frac{n}{2} \cos \frac{\theta}{2}
\end{bmatrix}
\]

or \(\epsilon \equiv \) eq (2) \(\equiv \) eq (3)

These equations are best represented by Figure 2a and b.

\[\text{Figure 2:}\]

6. EXAMPLE 2: GENERAL MOVEMENT OR A SCREW MOTION

Let’s apply two successive screw motions to a rigid body: the first one around the Oy axis of amplitude \(\theta_1 = \frac{\pi}{2}\) and of pitch \((p = \frac{2\pi}{3}, t = 4)\) followed by a second one around the Ox axis and of the same amplitude \(\theta_2 = \frac{\pi}{2}\) and same pitch \(p = 4\) corresponding to a translation of 1 unit along the two chosen axes:

\[
T_2 \cdot T_1 = \begin{bmatrix}
1 & 0 & 0 & 1 \\
0 & 0 & -1 & 0 \\
0 & 1 & 0 & 0 \\
0 & 0 & 0 & 1
\end{bmatrix}
\begin{bmatrix}
0 & 0 & 1 & 0 \\
0 & 1 & 0 & 1 \\
0 & 1 & 0 & 0 \\
0 & 0 & 0 & 1
\end{bmatrix} =
\begin{bmatrix}
0 & 0 & 1 & 1 \\
0 & 1 & 0 & 1 \\
1 & 0 & 0 & 1 \\
0 & 0 & 0 & 1
\end{bmatrix}
\]

The rotation part of the product corresponds to that of the precedent example of successive rotations \(R_1 = R_2\) with amplitude \(\theta = \frac{2\pi}{3}\) = 120 ° around the unit axis \(n = \frac{1}{\sqrt{3}}\); its translation part being \(t = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}\)

We can find its pitch \(p = \frac{2\pi}{\theta} (n \cdot t) = \frac{2\pi}{\frac{\pi}{3}} \begin{bmatrix} 1 \\ 1 \\ 0 \end{bmatrix} = \frac{6}{\sqrt{3}} = 2\sqrt{3}\)

The axis of rotation will keep its same original direction \(n = \frac{1}{\sqrt{3}}\); it will go through a new centre \(C\) given by the shifting vector \(u\) which could be found by the linear equations system: \((I - R) u = t - \frac{6}{\sqrt{3}} n\)

\[
\begin{bmatrix}
1 & 0 & -1 \\
-1 & 1 & 0 \\
0 & -1 & 1
\end{bmatrix}
\begin{bmatrix}
\frac{1}{\sqrt{3}} \\
\frac{1}{\sqrt{3}} \\
\frac{1}{\sqrt{3}}
\end{bmatrix}
= \begin{bmatrix}
\frac{1}{3} \\
\frac{1}{3} \\
\frac{1}{3}
\end{bmatrix}
\]

The vector translation \(T\) or \(t\) of the movement \(\begin{bmatrix} 1 \\ 0 \\ 1 \end{bmatrix}\)

is the sum of the two main perpendicular vectors \(T_1 + T_2\) such as \(T_1\) to be chosen parallel to \(n\) while the rest \(T_2\) is the translation vector part responsible for the shifting of the axis to its final position through the new center \(C\) as such we have:

\[
T_1 = \begin{bmatrix}
\frac{2}{3} \\
\frac{2}{3} \\
\frac{2}{3}
\end{bmatrix}
\text{and} \ T_2 = \begin{bmatrix}
\frac{1}{3} \\
\frac{1}{3} \\
\frac{1}{3}
\end{bmatrix}
\]

\(T_1\) being the translation part parallel to \(n\) while \(T_2\) being the perpendicular one.

The solutions to the system of linear equations are:

\[
u_x - u_z = \frac{1}{3}, \ -u_x + u_y = -\frac{2}{3}; \text{and} \ -u_y + u_z = \frac{1}{3} (5)
\]

Choosing the centre \(C\) to belong to the plane (y-z); \(u_x = 0\) or \((C_x = 0)\) would imply the two coordinates representing the point \(C\) intersection of the shifted axis \(n\) with the (y-z) plane to be:

\[C_y = -\frac{2}{3} \text{ and } C_z = -\frac{1}{3}\]

For the (z-x) plane; \(u_y = 0\) or \((C_y = 0)\); \(C_z = \frac{1}{3}\) and \(C_x = \frac{2}{3}\).

And finally considering the (x-y) plane; \(u_z = 0\) or \((C_z = 0)\); \(C_x = \frac{1}{3}\) and \(C_y = -\frac{1}{3}\).

So that to confirm these results; we can finally check the following conjugation matrices:
\[
\begin{pmatrix}
0 & 0 & 1 & 0 \\
1 & 0 & 0 & \frac{2}{3} \\
0 & 1 & 0 & \frac{2}{3} \\
0 & 0 & 0 & 1
\end{pmatrix}
\begin{pmatrix}
0 & 0 & 1 & \frac{2}{3} \\
1 & 0 & 0 & \frac{2}{3} \\
0 & 1 & 0 & \frac{2}{3} \\
0 & 0 & 0 & 1
\end{pmatrix}
= \\
\begin{pmatrix}
0 & 0 & 1 & \frac{1}{3} \\
1 & 0 & 0 & \frac{1}{3} \\
0 & 1 & 0 & \frac{1}{3} \\
0 & 0 & 0 & 1
\end{pmatrix}
\equiv (4)
\]

Or,
\[
\begin{pmatrix}
0 & 0 & 1 & \frac{2}{3} \\
1 & 0 & 0 & 0 \\
0 & 1 & 0 & \frac{2}{3} \\
0 & 0 & 0 & 1
\end{pmatrix}
\begin{pmatrix}
0 & 0 & 1 & \frac{2}{3} \\
1 & 0 & 0 & 0 \\
0 & 1 & 0 & \frac{2}{3} \\
0 & 0 & 0 & 1
\end{pmatrix}
= \\
\begin{pmatrix}
0 & 0 & 1 & \frac{1}{3} \\
1 & 0 & 0 & \frac{1}{3} \\
0 & 1 & 0 & \frac{1}{3} \\
0 & 0 & 0 & 1
\end{pmatrix}
\equiv (4)
\]

Or finally,
\[
\begin{pmatrix}
0 & 0 & 1 & \frac{1}{3} \\
1 & 0 & 0 & \frac{1}{3} \\
0 & 1 & 0 & \frac{1}{3} \\
0 & 0 & 0 & 1
\end{pmatrix}
\begin{pmatrix}
0 & 0 & 1 & \frac{1}{3} \\
1 & 0 & 0 & \frac{1}{3} \\
0 & 1 & 0 & \frac{1}{3} \\
0 & 0 & 0 & 1
\end{pmatrix}
= \\
\begin{pmatrix}
0 & 0 & 1 & -\frac{1}{3} \\
1 & 0 & 0 & \frac{1}{3} \\
0 & 1 & 0 & \frac{1}{3} \\
0 & 0 & 0 & 1
\end{pmatrix}
\equiv (4)
\]

\[q_2, q_1 = (q_2 + \frac{\varepsilon}{2} q_{x2}), (q_2 + \frac{\varepsilon}{2} q_{x2}) = q_2, q_1 + \frac{\varepsilon}{2} (q_2, q_{x2} + q_{x2}, q_1) = \left(\frac{\sqrt{2}}{2}, \frac{\sqrt{2}}{2}, 0, 0\right), \left(\frac{\sqrt{2}}{2}, 0, \frac{\sqrt{2}}{2}, 0\right), \left(\frac{\sqrt{2}}{2}, 0, \frac{\sqrt{2}}{2}, 0\right), \left(\frac{\sqrt{2}}{2}, 0, \frac{\sqrt{2}}{2}, 0\right), \left(\frac{\sqrt{2}}{2}, 0, \frac{\sqrt{2}}{2}, 0\right), \left(\frac{\sqrt{2}}{2}, 0, \frac{\sqrt{2}}{2}, 0\right)
\]

Another way of doing it: We could get this same result starting from the (4x4) rigid transformation eq (4) matrix defined before: A rotation of amplitude \(\theta = \frac{2\pi}{3} = 120^\circ\) around the unit axis \(n = \frac{1}{\sqrt{3}}[1, 1, 1]\) followed by a translation \(t = \begin{pmatrix}1 \\ 1 \end{pmatrix}\) such that:
\[
q = q + \varepsilon q_e = q_R + \frac{\varepsilon}{2}(t_x i + t_y j + t_z k)q_R = R + \varepsilon \frac{TR}{2} = \left(\frac{\sqrt{2}}{2}, \frac{\sqrt{2}}{2}, 0, \frac{\sqrt{2}}{2}\right) + \left(0, 0, 0, \frac{\sqrt{2}}{2}\right) = \left(\frac{\sqrt{2}}{2}, \frac{\sqrt{2}}{2}, 0, \frac{\sqrt{2}}{2}\right) + \left(0, 0, 0, \frac{\sqrt{2}}{2}\right) + \left(\frac{\sqrt{2}}{2}, \frac{\sqrt{2}}{2}, 0, \frac{\sqrt{2}}{2}\right) \equiv \left(\frac{\sqrt{2}}{2}, \frac{\sqrt{2}}{2}, 0, \frac{\sqrt{2}}{2}\right) + \left(\frac{\sqrt{2}}{2}, \frac{\sqrt{2}}{2}, 0, \frac{\sqrt{2}}{2}\right)
\]

At this stage we know the complete integrality of informations concerning this movement thanks to our magic and powerful calculated dual quaternion: The rotation part, as seen before, having amplitude \(\theta = \frac{2\pi}{3} = 120^\circ\) around the unit axis \(n, n = \frac{1}{\sqrt{3}}[1, 1, 1]\); the dual part will provide us gratefully with the translation along the axis of rotation; using eq (2): \(\varepsilon \left\{ -\frac{d}{2} \sin \frac{\pi}{2} \theta, \frac{m \sin \theta}{2} + \frac{d}{2} n \cos \frac{\theta}{2}\right\} = \left(\frac{\sqrt{2}}{2}, -\frac{1}{2}, 0, 0\right)\) implying that \(d = \frac{2}{\sqrt{3}} = 2\sqrt{3}\) and pitch \(p = 2\sqrt{3}\).

We can also have the vector part: \(\left[m \sin \theta + \frac{d}{2} n \cos \frac{\theta}{2}\right] = \left(0, 0, \frac{1}{2}\right)\) which implies:
\[
m_x \frac{\sqrt{3}}{2} + \frac{\sqrt{3}}{6} \frac{\sqrt{3}}{2} = m_x \frac{\sqrt{3}}{2} + \frac{1}{6} = 0
\]
\[
m_y \frac{\sqrt{3}}{2} + \frac{\sqrt{3}}{6} \frac{\sqrt{3}}{2} = m_y \frac{\sqrt{3}}{2} + \frac{1}{6} = 0
\]
\[
m_z \frac{\sqrt{3}}{2} + \frac{\sqrt{3}}{6} \frac{\sqrt{3}}{2} = m_z \frac{\sqrt{3}}{2} + \frac{1}{6} = 0
\]

We can then deduce the vector moment \(m = \left[\begin{smallmatrix} \frac{-1}{3\sqrt{3}} \\ \frac{-1}{3\sqrt{3}} \\ \frac{1}{3\sqrt{3}} \end{smallmatrix}\right]\)

Finally we can have the right position of the shifted axis \(u\) that have the same direction as the rotation axis.
\[ n \text{ by defining the coordinates } u_x, u_y \text{ and } u_z \text{ of a point or a center } C \text{ belonging to it so that: } m = u \land n \]

Or \[ \begin{cases} \frac{-1}{3\sqrt{3}} & u_x \\ \frac{-1}{3\sqrt{3}} & u_y \\ \frac{2}{3\sqrt{3}} & u_z \end{cases} \begin{bmatrix} 1 \\ \frac{1}{\sqrt{3}} \end{bmatrix} = \begin{bmatrix} u_y - u_z \\ u_x - u_y \end{bmatrix} \text{ implying that: } u_y - u_z = \frac{-1}{3}; u_z - u_x = \frac{-1}{3} \text{ and } u_x - u_y = \frac{2}{3} \]

Which confirm the same obtained results eq (5) using the (4x4) rigid transformation matrix:

\[ u_x - u_z = \frac{1}{3}; -u_x + u_y = -\frac{2}{3}; \text{ and } -u_y + u_z = \frac{1}{3} \text{ (5)} \]

8. THE 3R PLANAR MANIPULATOR

The planar manipulator is constituted of the three successive links (arms) (Figure 3) of lengths \( a_1, a_2 \) and \( a_3 \) that are rotating about their different axes, parallel to Oz: The vector dual quaternion \( q_v \), representing the coordinates of the end effector, to be manipulated is \( q_v = 1 + \varepsilon (a_1 + a_2 + a_3, 0, 0) \)

The result of the three manipulations would be elegantly given by the product of the dual quaternions in the following order; this could be done either by the operation:

\[ q_v' = q_1 q_2 q_3 q_v q_3^* q_2^* q_1^* = (q_1 q_2 q_3)(q_v)(q_3^* q_2^* q_1^*) = \]

Or the operation \[ q_1 q_2 q_3 q_v q_3^* q_2^* q_1^* = q_1 (q_2 (q_3 q_v q_1^*) q_2^*) q_1^* \text{ (7)} \]

Let us begin by the first operation concerning the rotation of amplitude \( \theta_3 \) around the axis \( C_z \) of the third link \( a_3 \) represented by the central DQ to somehow deploy outward these multiplications: \( q_3 q_v q_3^* \)

To find the dual quaternion \( q_3 \) we will need the conjugation technique given by the treble multiplication \( T R T^{-1} = (T)(R)(T^{-1}) \): Since the rotation is around \( C_z \); The coordinates of \( C_3 \) are: \( (a_1 + a_2, 0, 0) \), the rotation is: \( R_3 = \begin{pmatrix} \cos \frac{\theta_3}{2}, \sin \frac{\theta_3}{2} \\ 0, 0, 1 \end{pmatrix} \) and the translation is: \( T_3 = 1 + \frac{\varepsilon}{2} (a_1 + a_2, 0, 0) \) or \( 1 + \frac{\varepsilon}{2} t_3 \)

\[ q_3 = T_3 R_3 T_3^{-1} = [1 + \frac{\varepsilon}{2} (a_1 + a_2, 0, 0)] \]

\[ ([\cos \frac{\theta_3}{2}, \sin \frac{\theta_3}{2} (0, 0, 1)] [1 - \frac{\varepsilon}{2} (a_1 + a_2, 0, 0)] \text{ or} \]

\[ [1 + \frac{\varepsilon}{2} t_3] [R_3] [1 - \frac{\varepsilon}{2} t_3] = [1 + \frac{\varepsilon}{2} t_3] [R_3 - \frac{\varepsilon}{2} R_3 t_3] = \]

\[ R_3 + \frac{\varepsilon}{2} (t_3 R_3 - R_3 t_3) \text{ or similarly; } \]

\[ R_3 + \frac{\varepsilon}{2} t_3 R_3 [1 - \frac{\varepsilon}{2} t_3] = R_3 + \frac{\varepsilon}{2} (t_3 R_3 - R_3 t_3) \]

To simplify the writings, let’s adopt \( C_3 \) and \( S_3 \) in place of \( \cos (\theta_3 / 2) \) and \( \sin (\theta_3 / 2) \) respectively:

\[ R_3 = \begin{pmatrix} C_{\theta_3}, 0, 0, S_{\theta_3} \end{pmatrix}, \quad t_3 R_3 = [0, (a_1 + a_2), 0, 0] \]

and \( R_3 t_3 = [C_{\theta_3}, 0, 0, S_{\theta_3}] [0, (a_1 + a_2), 0, 0] = [0, C_3(a_1 + a_2), S_3(a_1 + a_2), 0] \)

\[ q_3 = R_3 + \frac{\varepsilon}{2} (t_3 R_3 - R_3 t_3) = [C_{\theta_3}, 0, 0, S_{\theta_3}] + \frac{\varepsilon}{2} [0, 0, -2S_3(a_1 + a_2), 0] \text{ (8)} \]

Important note: As a matter of fact nor do need the Denavit-Hartenberg parameters ‘avoiding to be lost in the maze of numerous parameters choices’ neither do we need the conjugation technique;

We already have this dual quaternion \( q_3 \) from definition (2);\n
\[ q_3 \equiv T = \begin{pmatrix} \cos \theta, \sin \theta \end{pmatrix} + \begin{pmatrix} 0, \sqrt{2} \end{pmatrix} \varepsilon \begin{pmatrix} d \sin \phi, \sqrt{2} \sin \phi n \end{pmatrix} \]

Noting that this manipulation concerns the shifting of the \( Oz \) axis the quantity \( (a_1 + a_2) \) along the \( x \)-direction and since there is no translation along the shifted axis \( C_z \), so replacing \( d = 0 \) in eq (2), will give:

\[ q_3 = \begin{pmatrix} \cos \frac{\theta}{2}, \sin \frac{\theta}{2} \phi \end{pmatrix} \equiv [C_{\theta}, (0, 0, S_{\theta})] + \varepsilon \begin{pmatrix} 0, \sqrt{2}, \{m \sin \theta + \frac{2}{\sqrt{3}} n \cos \theta \} \end{pmatrix} \]

Finding the moment \( m \) of the \( (a_1 + a_2) \) with respect to the axis of rotation \( k \) will give us \( q_3; \)

\[ m = \begin{cases} a_1 + a_2 & 0 \\ 0 & 1 \end{cases} \quad \varepsilon \begin{pmatrix} 0, 0, -2S_3(a_1 + a_2) \end{pmatrix} \]

\[ \text{and thus } q_3 = [C_{\theta}, (0, 0, S_{\theta})] + \varepsilon \begin{pmatrix} 0, 0, -(a_1 + a_2) S_3, 0 \end{pmatrix} \text{ (9)} \]

Exactly the same result but elegantly and with very much less hassle!

We can then have:

![Figure 3: Manipulator RRR and its home position.](image-url)
\[ q_3 q_2 q_1 = [(c_3, 0, 0, s_3) + \varepsilon (0, -s_3(a_1 + a_2), 0)] [(1 + \varepsilon ((a_1 + a_2 + a_3), 0, 0)) [(c_3, 0, 0, -s_3) + \varepsilon (0, -s_3(a_1 + a_2), 0)]] \\

Performing properly the products and using the fundamental trigonometric properties we can find the vector result of this first transformation: 
\[ 1 + \varepsilon (a_1 + a_2 + a_3 \cos \theta_2, a_3 \sin \theta_3, 0) \]

Finally, the first movement of our manipulator can be represented by the Figure 4:

For the second rotation of angle \( \theta_2 \) concerning the second link \( a_2 \), about the axis \( C_2z \) and using the same procedure we can write the dual quaternion \( q_2 = [(c_2, 0, 0, s_2) + \varepsilon (0, -s_2 a_1, 0)] \)

To be applied to the found precedent result vector:
\[ 1 + \varepsilon (a_1 + a_2 + a_3 \cos \theta_3, a_3 \sin \theta_3, 0) = 1 + \varepsilon (x, y, 0) \]

We will have:
\[ [(c_2, 0, 0, s_2) + \varepsilon (0, -s_2 a_1, 0)][(1 + \varepsilon (x, y, 0)) \] 
\[ [(c_2, 0, 0, s_2) + \varepsilon (0, -s_2 a_1, 0)] = 1 + \varepsilon ((1 - \cos \theta_2) a_1 - \sin \theta_2 y + \cos \theta_2 x, -a_1 \sin \theta_2 + \cos \theta_2 y + \sin \theta_2 x, 0) \]

And finally giving \( x \) and \( y \) their precedent values:
\[ (x, y, 0) = (a_1 + a_2 + a_3 \cos \theta_3, a_3 \sin \theta_3, 0) \]

The coordinates of the resulting vector after the transformation will be:
\[ (x, y, z) = (a_1 + a_2 \cos \theta_2 + a_3 \cos (\theta_2 + \theta_3), a_2 \sin \theta_2 + a_3 \sin (\theta_2 + \theta_3), 0) \]; See Figure 5:

The last and final movement is a pure rotation \( (\theta_1) \) around the axis \( Oz \) applied to the precedent result vector:
\[ q_1 = (c_1, 0, 0, s_1)[1 + \varepsilon (x, y, 0)] (c_1, 0, 0, s_1) = 1 + \varepsilon (-\sin \theta_1 x + \cos \theta_1 y + \sin \theta_1 x, 0) \]

Finally and replacing \( x \) et \( y \) by their precedent values:
\[ (x, y, 0) = (a_1 + a_2 \cos \theta_2 + a_3 \cos (\theta_2 + \theta_3), a_2 \sin \theta_2 + a_3 \sin (\theta_2 + \theta_3), 0) \]

This will give us the final position of the vector (representing the end effector) result after the three successive rotations:
\[ x = a_1 \cos \theta_1 + a_2 \cos (\theta_1 + \theta_2) + a_3 \cos (\theta_1 + \theta_2 + \theta_3), \]
\[ y = a_1 \sin \theta_1 + a_2 \sin (\theta_1 + \theta_2) + a_3 \sin (\theta_1 + \theta_2 + \theta_3) \]
\[ z = 0 \]

Thus, one can easily confirm all the results obtained for this example using either the D.Q method or the classical Denavit and Hartenberg parameters method (See 10.3.2.) or the matrix conjugation technique \( T. M . T^{-1} \).
Nevertheless it is nothing but pure common sense to
find these direct 3R manipulator positions by drawing the
successive positions of the links (components) of
the RRR manipulator.

9. CONCLUSION

We hope that the reader should not get us wrong:
We never pretend that the D-H parameters method is
wrong or obsolete and that it should be a thing of the
past; recognizing that this important classical method
was the precursor that enlightened the path to modern
robotics; we only say and insist that there exist through
the DQ parameters another short, free of singularities
and easy to work with method, when dealing with robot
direct kinematics. On the light of the obtained results
one has to say that the most perfect (not suffering
singularities of any kind), easiest and rapid way to
perform a 3D rigid transformation of any sort is to use
the dual quaternion that characterises the movement.
Most of all we are free to use the 3D space, being sure
that no loss of degree of freedom or guimball lock of
any sort can never happen. Using a D-H parameters
method or any of its counterparts means a choice of
different sort of embracing and somehow awkward
three axes frames to be created and then allocated to
each arm/ link; ‘providing’ our robot or mecanism with
different direction axes and angles with very much
complicated choice of signs (concerning the directions
and the angles alike) to be chosen subject to some
rules depending on the chosen method and model of
robot.

Choosing to use dual quaternions we only need to
know the constants or values that concern the
construction geometry of a given or chosen robot
(directions of rotations, distances, lengths of links...) to
evaluate its kinematics without any threat to be lost in
the maze or jungle of choices. Most of all, it will prevent
us from using the only other existing method, or one of
its options, which is that of the Denavit and Hartenberg
parameters which mainly consists of: 1) Choosing 3D
frames attached to each link upon certain conditions
/conventions, 2) Schematic of the numbering of bodies
and joints in a robotic manipulator, following the
convention for attaching reference frames to the bodies,
this will help to create: 3) A table for exact definition of
the four parameters, $a_i$, $d_i$, and $q$, that locate one
frame relative to another, 4) The (4x4) rigid
transformation matrix that will have the given form: $T_{i-1}^{i}$. (See 11.3)

This chapter provided a taste of the potential
advantages of dual-quaternions, and one can only
imagine the further future possibilities that they can offer. For example, there is a deeper investigation of
the mathematical properties of dual-quaternions (e.g.,
zero divisions). There is also the concept of
dual-dual-quaternions (i.e., dual numbers within dual
numbers) and calculus for multi-parametric objects for
the reader to pursue if he desires.

This Dual Quaternions Kinematics method could be
easily generalized to all kinds of existing or/and future
robots providing their given general geometric
characteristics to be known.

We should emphasize on the fact that Matlab
software was used, throughout this chapter and
whenever necessary, concerning all kinds of products
or multiplication of quaternions or rigid transformation
matrices.

Finally we hope all efforts should be conjugated to
create a common MATLAB platform to be used for the
manipulation of Quaternions and / or Dual Quaternions
as well as conversions from or into 3D or 4D rigid body
matrices.

10. APPENDIX

10.1. Quaternion-Matlab Implementation Class

>> % See paragraph 3; Example 1: Rotations
represented by Quaternions >> % A first rotation of
angle π/2 around the x-axis ,q1 , followed by a rotation
of angle π/2 around the y-axis , q2 will result in a
rotation given by the product n1 = q2.q1 :

q2 = [cos(pi/4) sin(pi/4) 0 0 ];
q1 = [0 0 1; 0 1 0 ; 1 0 0 ];

n1 = quatmultiply (q2,q1)

>> % If the order is inversed the result will be given by
the quaternion n2 = q1.q2

n2 = [cos(pi/4) 0 sin(pi/4) 0 ];

>> n1 = quatmultiply (q2,q1)

>> % A first rotation of

>> % Using 3*3 matrices; if the rotation R1 is
performed first the rotation product is R2*R1:

R1 = [1 0 0; 0 1 0; 0 0 -1];
R2 = [0 0 1; 0 1 0; -1 0 0 ];

prod1 = R2*R1

prod1 =
0 1 0
0 0 -1
-1 0 0

>> % if the order is inversed the multiplication will be
R1*R2:

prod2 = R1*R2

prod2 =
0 0 1
1 0 0
0 1 0
10.2. Quaternions and Dual Quaternions (DQ)

10.2.1. Quaternions or Rotation Representation

Quaternions were first discovered and described by the Irish mathematician Sir Rowan Hamilton in 1843. Indeed quaternion’s representation and axis-angle representation are very similar.

Both are represented by the four dimensional vectors. Quaternions also implicitly represent the rotation of a rigid body about an axis. It also provides better means of key frame interpolation and doesn’t suffer from singularity problems.

The definition of a quaternion can be given as \((s, \mathbf{m})\) or \((s, q_x, q_y, q_z)\) where \(\mathbf{m}\) is a 3D vector, so quaternions are like imaginary (complex) numbers with the real scalar part \(s\) and the imaginary vector part \(\mathbf{m}\).

Thus it can be also written as: \(s + q_x i + q_y j + q_z k\).

There are conversion methods between quaternions, axis-angle and rotation matrix.

Common operations such as addition, inner product etc can be defined over quaternions.

Given the definition of \(q_1\) and \(q_2\):

\[
q_1 = s_1 + q_{x_1} i + q_{y_1} j + q_{z_1} k \quad \text{or} \quad q_1 = (s_1, \mathbf{m}_1)
\]

\[
q_2 = s_2 + q_{x_2} i + q_{y_2} j + q_{z_2} k \quad \text{or} \quad q_2 = (s_2, \mathbf{m}_2)
\]

Addition operation is defined as:

\[
q_1 + q_2 = (s_1 + s_2, \mathbf{m}_1 + \mathbf{m}_2) = (s_1 + s_2) + (q_{x_1} + q_{x_2})i + (q_{y_1} + q_{y_2})j + (q_{z_1} + q_{z_2})k
\]

dot (scalar, inner): product operation( . ) as:

\[
q_1 \cdot q_2 = s_1 s_2 + \mathbf{m}_1 \cdot \mathbf{m}_2
\]

Quaternion multiplication is non commutative, but it is associative.

Multiplication identity element is defined as : \((1, (0, 0, 0))\)

We can also perform the multiplication in the imaginary number domain using the definitions:

\[
i^2 = j^2 = k^2 = -1; \quad i, j = k, j, k = i, k, i = j;
\]

\[
i, j = -k, k, j = i, i, k = -j
\]

Equations (A1) to (A12) state the definitions, rules and properties of dual quaternion algebra.

Quaternion multiplication \((\otimes)\) is defined as:

\[
q_1 \otimes q_2 = (s_1, s_2 - \mathbf{m}_1 \cdot \mathbf{m}_2, s_1 \cdot \mathbf{m}_2 + s_2 \cdot \mathbf{m}_1 + \mathbf{m}_1 \wedge \mathbf{m}_2)
\]  \hspace{1cm} (10)

Each quaternion has a conjugate \(q^*\) and an inverse \((q \neq 0)\) \(q^{-1}\) where \(|q|^2 = s^2 + q_x^2 + q_y^2 + q_z^2 = q \otimes q^* = q^* \otimes q\)

and an inverse \(q^{-1} = (\frac{1}{|q|^2})q^*\).

Rotations are defined by unit quaternions. Unit quaternions must satisfy \(|q| = 1\). Since multiplication of two unit quaternions will be a unit quaternion, \(N\) rotations can be combined into one unit quaternion \(q_R = q_{R_1} \otimes q_{R_2} \otimes q_{R_3} \ldots q_{R_N}\).

It is also possible to rotate a vector directly by using quaternion multiplication. To do this, we must define a 3D vector \(\mathbf{v} = (v_x, v_y, v_z)\) that we want to rotate in quaternion definition as \(q_v = (0, \mathbf{v}) = 0 + v_xi + v_yj + v_zk\).

The rotated vector \(\mathbf{v}' = (v'_x, v'_y, v'_z)\) can be defined as:

\[
q_v = q \otimes q_v \otimes q_v^{-1} = q \otimes q_v \otimes q^*
\]  \hspace{1cm} (12)

And, assuming another quaternion rotation \(p\), two rotations can be applied to the vector \(\mathbf{v}\) such as:

\[
q_v = p \otimes (q \otimes q_v \otimes q_v^{-1}) \otimes p^{-1} = (p \otimes q_v \otimes (q^{-1} \otimes p^{-1})) = C \otimes q_v \otimes C^{-1}
\]  \hspace{1cm} (13)

providing that quaternion \(C = (p \otimes q)\) is a combination of the precedent quaternions \(q\) and \(p\).

The equation implies that vector \(\mathbf{v}\) is first rotated by the rotation represented by \(q\) followed by the rotation \(p\).

A quaternion \(q\) that defines a rotation about (around) the axis \(\mathbf{n}\) denoted by the unit vector \((n_x, n_y, n_z)\) of an angle \(\theta\) could be written as:

\[
q = \cos \frac{\theta}{2} + \sin \frac{\theta}{2} (n_x i + n_y j + n_z k)
\]  \hspace{1cm} (14)

This same quaternion represents a rotation of amplitude \((-\theta)\) around the opposite axis \((-\mathbf{n})\).

10.2.2. Dual Quaternions

Dual Quaternions (DQ) were proposed by William Kingdom Clifford in 1873. They are an extension of quaternions. They represent both rotations and translations whose composition is defined as a rigid transformation.

They are represented by the following eight dimensional vector:

\[
\mathbf{q} = (s, \mathbf{m}) = (s, q_x, q_y, q_z, q_{\varepsilon x}, q_{\varepsilon y}, q_{\varepsilon z}, q_{\varepsilon x}, q_{\varepsilon y}, q_{\varepsilon z})
\]  \hspace{1cm} (15)

Such that: \(\mathbf{q} = q + \varepsilon q_e = s + q_x i + q_y j + q_z k + \varepsilon (q_{\varepsilon x} + q_{\varepsilon x} + q_{\varepsilon y} + q_{\varepsilon z})\)
Dual quaternion multiplication is defined by:
\[
\tilde{q} \otimes \tilde{q}_2 = q_1 \otimes q_2 + \varepsilon \left( q_1 \otimes q_2 + q_1 \varepsilon \otimes q_2 \right)
\]
(16)

With \( \varepsilon^2 = 0 \); \( \varepsilon \) being the second order nilpotent dual factor.

The dual conjugate (analogous to complex conjugate) is denoted by:
\[
\bar{q} = q - \varepsilon q \varepsilon
\]
(17)

This conjugate operator can lead to the definition of the inverse of \( \bar{q} \) which is:
\[
q^{-1} = \frac{1}{q} = \frac{\bar{q}}{q^2} = \frac{1}{q} - \varepsilon \frac{q \varepsilon}{q^2} \quad \text{which means that a pure dual number (i.e:} q = 0 \text{)} \quad \text{does not have an inverse.}
\]

\[
\bar{q} = - q \otimes q^{-1} = (q + \varepsilon q \varepsilon) \left( \frac{1}{q} - \varepsilon \frac{q \varepsilon}{q^2} \right) = \frac{q}{q} - \varepsilon \frac{q \varepsilon}{q^2} + \frac{\varepsilon q}{q} - \varepsilon \frac{q \varepsilon}{q^2} + \varepsilon q \varepsilon = 1 - 0 = 1
\]

A second conjugation operator is defined for DQs. It is the classical quaternion conjugation and is denoted by: \( q^* = q^* - \varepsilon q \varepsilon \).

Where conjugation of dual and non-dual quaternion parts satisfies eq (11).

Combining these two conjugation operators will lead to the formalization of DQ transformation on 3D points. Use of both conjugations on \( q \) can be denoted \( \bar{q}^* \). Using definitions (11), (15) and (17) we finally have:
\[
\bar{q}^* = (s, -q_x, -q_y, -q_z, -q_{\varepsilon x}, q_{\varepsilon y}, q_{\varepsilon z})
\]
(18)

It is well known that we can use dual quaternions to represent a general transformation subject to the following constraints:

The DQ screw motion operator \( q^* = (q, q_x) \) must be of unit magnitude: \( |q| = (q + \varepsilon q \varepsilon)^2 = 1 \)

This requirement means two distinct conditions or constraints:
\[
s^2 + q_x^2 + q_\varepsilon^2 + q_{\varepsilon x}^2 = 1 \quad \text{and}
\]
\[
s q_x + q_\varepsilon q_{\varepsilon x} + q_x q_{\varepsilon y} + q_\varepsilon q_{\varepsilon z} = 0
\]
(19)

Which imposed on the eight (8) parameters of a general DQ, effectively reduce the number of degree of freedom (8 – 2) = 6; equivalent to the degree of freedom of any free rigid body in 3-D space

10.2.3. Dual Quaternions or General 3D Rigid Transformation Representation

While equation (14) defines completely and unambiguously (without any singularity like guimbal lock and other loss of degree of freedom) all 3D rotations in the physical space, dual quaternions can represent translations;

A DQ defined as: \( \hat{q}_T = 1 + \frac{\varepsilon}{2} (t_x i + t_y j + t_z k) \) corresponds to the translation vector \( \hat{T} = (t_x, t_y, t_z) \)

Which could symbolically be noted \( T \); so \( \hat{q}_T = 1 + \varepsilon \frac{T}{2} \)

The translation \( T \) on the vector \( \hat{v} \) can be computed by: \( \hat{q}_v \otimes \hat{q}_v \otimes \hat{q}_T \)

So fortunately using def (A9), we have: \( \bar{q}_v^* = q_v^* = 1 + \varepsilon \frac{T}{2} \) then \( \hat{q}_v^* = \hat{q}_T \otimes \hat{q}_v \otimes \hat{q}_T = \hat{q}_T \otimes \hat{q}_v \otimes \hat{q}_T = [1 + \varepsilon \left( v_x i + v_y j + v_z k \right)] \otimes [1 + \varepsilon \left( v_x i + v_y j + v_z k \right)] \otimes [1 + \varepsilon \left( v_x i + v_y j + v_z k \right)] = 1 + \varepsilon \left( v_x i + (v_y + t_y) j + (v_z + t_z) k \right) \)

Which correspond to the transformed vector: \( \hat{v'} = (v_x + t_x) i + (v_y + t_y) j + (v_z + t_z) k \)

10.2.4. Combining Rotations and Translations

Assuming: \( \hat{q} \) and then \( \hat{\beta} \), two DQ transformations applied successively and in that order to a DQ position vector \( \hat{q}_v \); Their combined DQ transformation \( \hat{C} \) applied to \( \hat{q}_v \) gives:
\[
\hat{q}_v = \hat{C} \otimes (\hat{q} \otimes \hat{q} \otimes \hat{\beta}^*) \otimes \hat{\beta}^* = (\hat{C} \otimes \hat{q} \otimes (\hat{q}^* \otimes \hat{\beta}^*)) = \hat{C} \otimes \hat{q} \otimes \hat{C}^*
\]
(20)

It is very important to notice that the most inner transformation of the equation is applied first with an inside to outside manner.

In eq (20), \( \hat{q} \) is the first transformation followed by the second one \( \hat{\beta} \).

The successive composition or combination of unit DQ rotation \( \hat{q}_R = R \) followed by a unit DQ translation \( \hat{q}_T = 1 + \frac{\varepsilon}{2} (t_x i + t_y j + t_z k) \)

will give: \( \hat{q}_T \otimes \hat{q}_R = (1 + \frac{\varepsilon}{2} (t_x i + t_y j + t_z k)) \otimes \hat{q}_R = \hat{q}_R + \varepsilon \frac{T}{2} (t_x i + t_y j + t_z k) \otimes \hat{q}_R = R + \varepsilon \frac{T R}{2} \)

(21)

Its inverse being: \( (R + \varepsilon \frac{T R}{2})^{-1} = R^* - \frac{R^* T}{2} \)

If the translation is applied first:
\[
\hat{q}_R \otimes \hat{q}_T = \hat{q}_R \otimes (1 + \frac{\varepsilon}{2} (t_x i + t_y j + t_z k)) = \hat{q}_R + \frac{1}{2} (t_x i + t_y j + t_z k) \otimes \hat{q}_R = R + \varepsilon \frac{R T}{2}
\]

(22)

Its inverse being: \( (R + \varepsilon \frac{R T}{2})^{-1} = R^* - \frac{R^* T}{2} \)

10.2.5. Several Transformations

Suppose that the vector \( V \) in its dual quaternion form \( \hat{q}_v = 1 + \varepsilon v \) is under a sequence of rigid transformations represented by the dual quaternions \( \hat{q}_1, \hat{q}_2, \ldots, \hat{q}_n \). The resulting vector is encapsulated in the dual quaternion:
\[ 1 + \varepsilon v' = \bar{q}_n \otimes (\bar{q}_{n-1} \otimes \ldots \otimes (\bar{q}_1 \otimes (1 + \varepsilon v) \otimes \bar{q}^*_1) \otimes \ldots \otimes \bar{q}^*_n) \]
\[ = (\bar{q}_n \otimes \ldots \otimes \bar{q}_1) \otimes (1 + \varepsilon v) \otimes (\bar{q}^*_1 \otimes \ldots \otimes \bar{q}^*_n) \]  

1. \( 1 + \varepsilon v' = \bar{q} \otimes (1 + \varepsilon v) \otimes \bar{q}^* \).

2. Definition of the main axes of each segment: • If \( z_i \) and \( z_{i+1} \) do not intersect we choose \( x_i \) so as to be the parallel with the axis perpendicular to \( z_i \) and \( z_{i+1}. \) • If \( z_i \) and \( z_{i+1} \) are collinear, \( x_i \) is chosen in the plane perpendicular to \( z_{i+1}. \)

3. Fix the four geometric parameters: \( d_i, \theta_i, a_i, \alpha_i \) (see Figure 7) for each joint such as:

- \( d_i \) coordinate of the origin \( O_i \) on the axis \( z_{i+1}. \) For a slide \( d_i \) is a variable and for a hinge \( d_i \) is a constant.
- \( \theta_i \) is the angle obtained by screwing \( x_{i-1} \) to \( x_i \) around the axis \( z_{i+1}. \) For a slide \( q_i \) is a constant and for a hinge \( q_i \) is a variable.
- \( a_i \) is the distance between the axes \( z_i \) and \( z_{i+1} \) measured on the axis \( x_i \) negative from its origin up to the intersection with the axis \( z_{i+1}. \)
- \( \alpha_i \) is the angle between \( z_i \) et \( z_{i+1} \) obtained by screwing \( z_{i-1} \) to \( z_i \) around \( x_i. \)

Finally, the homogeneous DH displacement matrix \( [T^d_{i-1}] \) which binds together the rotation and the translation is formed. Its left upper part defines the rotation matrix \( R^d_{i-1} \) and on its right the translation vector

\[ d^i_{i-1} = \begin{bmatrix} R^d_{i-1} & d^i_{i-1} \end{bmatrix} \]  
\[ \cos \theta_i & -\cos \alpha_i \sin \theta_i & \sin \alpha_i \sin \theta_i \\ \sin \theta_i & \cos \alpha_i \cos \theta_i & -\sin \alpha_i \cos \theta_i \\ 0 & \sin \alpha_i & \cos \alpha_i \]  
\[ \begin{bmatrix} a_i \cos \theta_i \\ a_i \sin \theta_i \\ d_i \end{bmatrix} = \begin{bmatrix} a_i \cos \theta_i \\ a_i \sin \theta_i \\ d_i \end{bmatrix} \]

And finally the \((4x4)\) rigid transformation matrix will have the form:

Figure 7: Coordinate systems and parameters of Denavit and Hartenberg.
10.3.2. D-H kinematics of the Planar RRR Robot

Null values must be taken for the parameters: \( d_0 = 0 \) and \( a_i = 0 \) in Figure 7 and matrix (28) to give:

The appropriate transformations for the first three considered articulations are:

\[
T_0^1 = \begin{pmatrix}
\cos \theta_1 & -\cos \alpha_1 \sin \theta_1 & a_1 \cos \theta_1 \\
\sin \theta_1 & \cos \alpha_1 \cos \theta_1 & a_1 \sin \theta_1 \\
0 & -\sin \alpha_1 & \cos \alpha_1
\end{pmatrix}
\]

\[
T_0^2 = \frac{c}{a} \begin{pmatrix}
1 & 0 & 0 \\
0 & \cos \alpha_2 & \sin \alpha_2 \\
0 & -\sin \alpha_2 & \cos \alpha_2
\end{pmatrix}
\]

\[
T_0^3 = \frac{c}{a} \begin{pmatrix}
1 & 0 & 0 \\
0 & \cos \alpha_3 & \sin \alpha_3 \\
0 & -\sin \alpha_3 & \cos \alpha_3
\end{pmatrix}
\]

The last column represents the position coordinates \( x \) and \( y \) of the end effector:

\[
x = a_1 c_1 + a_2 c_2 + a_3 c_3; \quad y = a_1 s_1 + a_2 s_2 + a_3 s_3
\]

REFERENCES

[22] Whittaker ET. A Treatise on Analytical Dynamics of Particles and Rigid Bodies, 1904; p. 4, at Google Books


Received on 26-05-2018

Accepted on 20-06-2018

Published on 02-08-2018

DOI: http://dx.doi.org/10.12970/2308-8354.2018.06.02

© 2018 Gouasmi et al.; Licensee Synergy Publishers. This is an open access article licensed under the terms of the Creative Commons Attribution Non-Commercial License (http://creativecommons.org/licenses/by-nc/3.0/) which permits unrestricted, non-commercial use, distribution and reproduction in any medium, provided the work is properly cited.