Quaternion and Dual Quaternion in Rigid Body Motion

-- ME5701 Mathematics for Engineering Research

A0219282A Wei Kexin

CONTENTS

1 INTRODUCTION

Quaternion and Dual Quaternion, which are created in 19th century, now are more widely applied in physics, robotics. Based on a research paper of using quaternion and dual quaternion to control a satellite to move to the desired position[1] this report explains the utility of these 2 mathematical tools in the rigid body motion, which contains the application method for distinct classes of quaternion and dual quaternion in Matlab, the rotation of vector using quaternion, the transformation of reference frame depending on dual quaternion, and redone part of the results from [1].

2 LITERATURE REVIEW

Quaternion, $q = q_0 + q_1 i + q_2 j + q_3 k$, is first described by William Rowan Hamilton[2], by using 3 imaginary part i, j, k. A unit quaternion is also seen as a spatial rotation represented in a special operator way, thus, can replace the Euler angle rotation matrix, and simultaneously avoids the gimbal lock problem in the Euler parameters[3]. The gimbal lock will make the described object with motion pitch, roll and yaw to lose 1 degree of freedom, and forever lock it there. This will not be solved by simply adding more rotation axis, and sequentially affects the attitude representation.

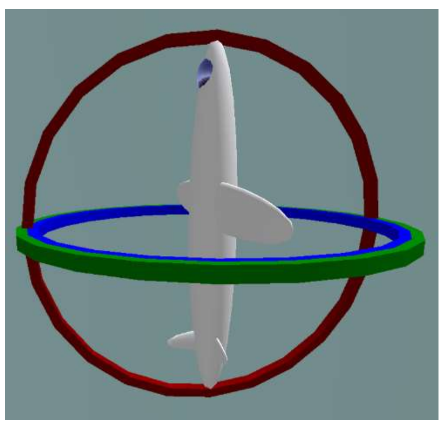

Fig 2.1: The gimbal lock[4]

The mathematical foundation of quaternion can be found in the [1], where the addition, multiplication, conjugate, norm and cross product will be frequently used in the rigid body motion. Furthermore, the relationship between the quaternion and rigid body rotation is revealed in [5], [6].

Dual quaternion[7] , $\hat{q} = q_r + \epsilon q_d$,combines quaternion and dual number and contributes an 8-dimensional tuple, with i, j, k and dual unit ϵ. Similar like the notation in [1], q_r and q_d are real and dual quaternion separately.

A unit dual quaternion can be seen as a rigid body transformation merging the rotation and translation[8]. To clarify, the norm of unit dual quaternion is defined as $\| \hat{q} \|_d^2 = \hat{q}\hat{q}^* = \hat{q}^*\hat{q} = \hat{q} \cdot \hat{q} = (q_r \cdot q_r) + \epsilon(2q_r \cdot q_d) = 1$.

3 IMPLEMENTATIONS

In this chapter, quaternion and dual quaternion will be used in Matlab to visualize the rigid body motion. All the mathematical fundamental applications for quaternion and dual quaternion are shown in "quaternion_demo.mlx" and "dual_quaternion_demo.mlx" and the pdf file with the corresponding names.

3.1 QUATERNION AS SPATIAL ROTATION

Quaternion can be used to describe a vector rotation in the same reference frame or vector seen from differed reference frames, which, known from 3D rigid body rotation, have same core but appear in inversed way.

First the example of the vector rotation in the same reference frame is demonstrated, where the mathematical proof is expressed in [5]. The equation is stated as

$$v_2 = q_{2/1} v_1 q_{2/1}^*$$

where the v_1 is the original vector in the quaternion form, $v_1 = (\vec{v}_1, 0)$. Here define $q_{2/1} = \cos(\theta/2) + \sin(\theta/2)\vec{n}$, where $\theta = 90°, \vec{n} = [-1, -1, 1]^T/\sqrt{3}$, to define the rotation from 1 to 2.

Since $q_{2/1}$ generates a rotation axis $\vec{n}$ with an angle θ, we can also directly use rotation matrix[9] for validation, $A(\theta, \vec{n}) = e^{\theta N}$, where N is the corresponding skew-symmetric matrix of $\vec{n}$.

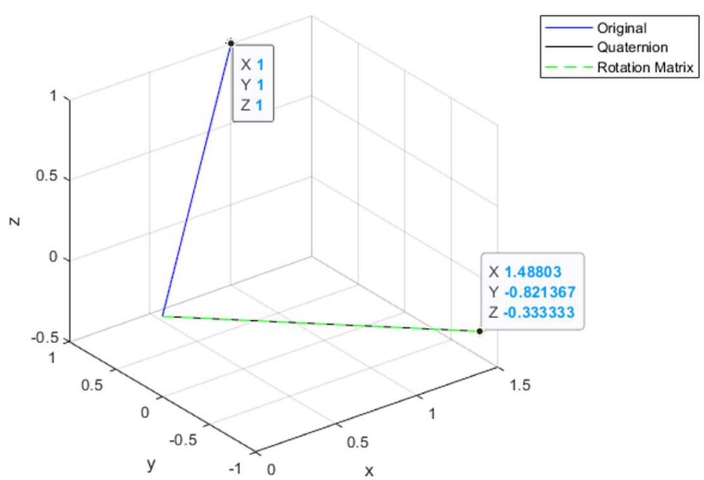

Fig 3.1: Vector rotation

The result indicates quaternion does represent the spatial rotation. Thus, for the vector seen from the different reference frame, the equation is $v^I = q_{B/I}\, v^B q_{B/I}^*$, or $v^B = q_{B/I}^*\, v^I q_{B/I}$. The example is produced from "quaternion_rotation.m".

3.2 DUAL QUATERNION AS TRANSFORMATION

Dual quaternion is the combination of translation and rotation, denoted as

$$\hat{q}_{B/I} = q_{B/I} + \frac{\epsilon}{2}\vec{t}_{B/I}q_{B/I}$$

where $\vec{t}_{B/I}$ is the displacement form reference frame I to B.

To perform this, a new class "ReferFrame" is used to store the 3D body motion's properties, including $r^I_{B/I}, w^I_{B/I}, v^I_{B/I}, a^I_{B/I}, \alpha^I_{B/I}, q_{B/I}$ so also the $\hat{q}_{B/I}$. This example is a circular motion to imitate the desired reference frame rotates around the target satellite, but changed with radius $R = 4$ m and $\vec{w}^I_{B/I} = [0,0,1]^T$ rad/sec. To verify the algorithm, the original trajectory is designed as $\vec{r}^I_{B/I} = [R\cos(wt), R\sin(wt), 0]^T$ m, but the real-time $\vec{r}^I_{B/I}(t)$ is numerically integrated by the $\vec{a}^I_{B/I} = [-Rw^2\cos(wt), -Rw^2\sin(wt), 0]^T$ m/s^2 and $\vec{\alpha}^I_{B/I} = [0,0,0]^T$m/s^2.

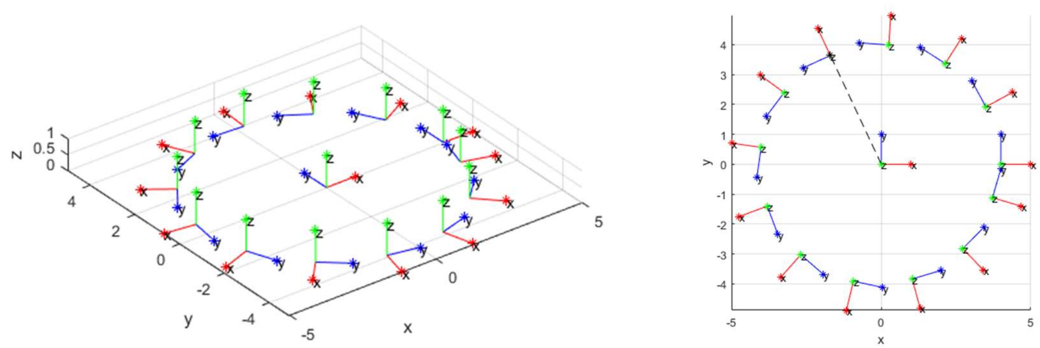

Fig 3.2: Circular Motion

However, since the velocity and displacement are recursive accumulated from $\vec{a}^I_{B/I}$, if the timestep is too large, the trajectory will have obvious deviation from the original designed one.

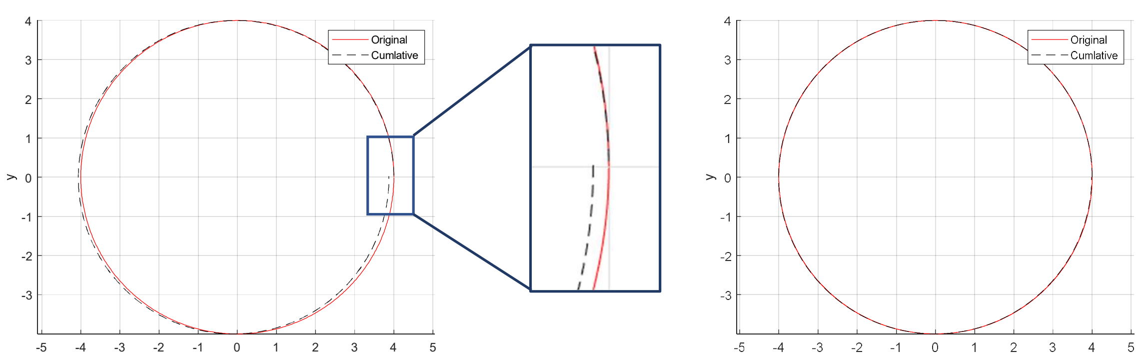

Fig 3.3: Trajectory deviation. 1) left: timestep = 0.01s; 2) right: timestep =0.001s

This can be fixed with smaller timestep, but also at the cost of computation speed. This result is created by "dual_quaternion_transformation.m".

To complexify the reference frame transformation, using an inclination $45°$, to an elliptical rotation, where the inertial reference frame is at the center of the ellipse and the semimajor $a = 2.66$ m, eccentricity $e = 0.74$.

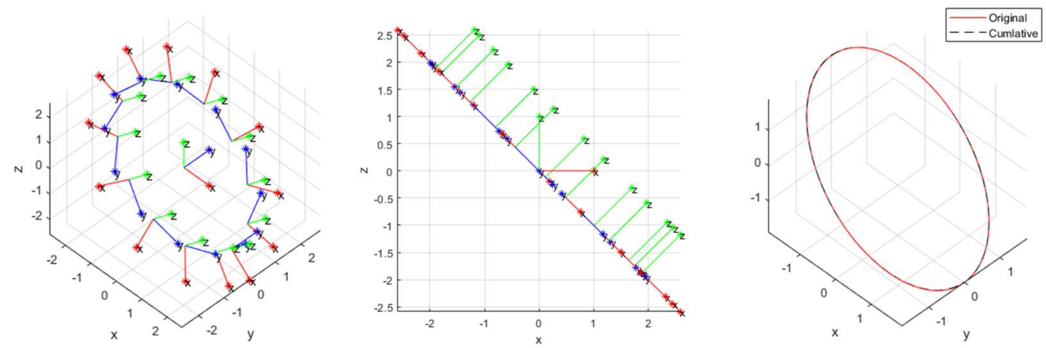

Fig 3.4: A simple elliptical orbit

To prove the cumulative trajectory created by angular acceleration, linear acceleration dual quaternion is correct, the original designed one is drawn as the reference. From the middle fig, the inclination is clearly seen as $45°$, and the last fig shows out the original trajectory and cumulative trajectory in the inertial reference frame. The result is generated by running "elliptical_motion.m".

3.3 MULTIPLE REFERENCE FRAMES

Combine the 2 motion from 3.2, we can obtain a more complex situation with desired reference frames, target reference frame and inertial reference frame, short for D, T and I.

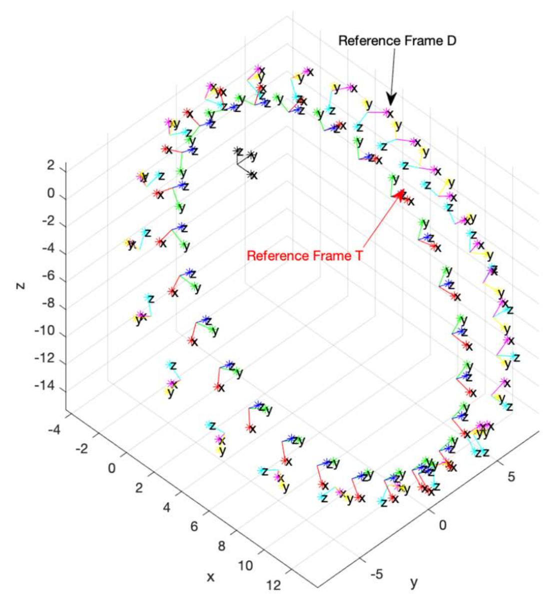

Fig 3.5: Combination of 2 motions

Reference frame D does circular rotation around T yet in the YZ plane and reference frame T rotates in the elliptical orbit around the center of the ellipse, which has an inclination of 45° respect to the I.

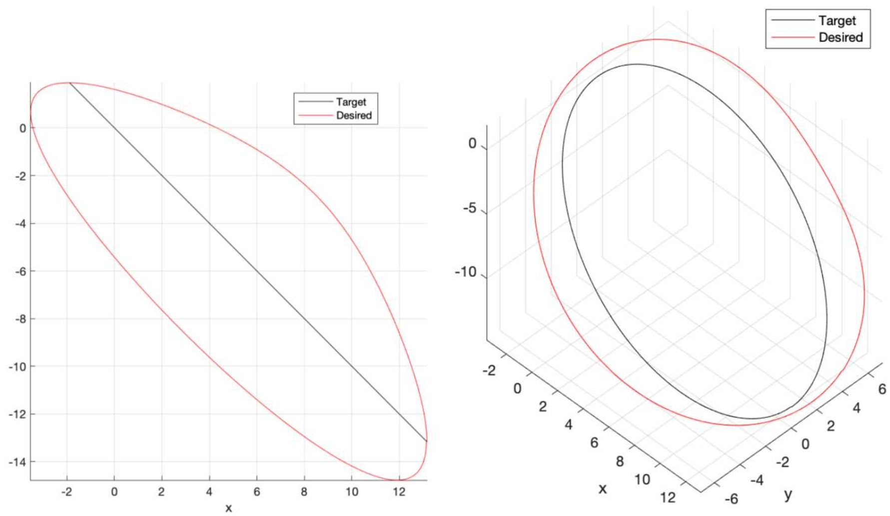

Fig 3.6: Trajectory of reference frames D and T

Even so this is bit of hard to prove the reference frames rotates rightly, to do so, we can arbitrarily choose a time t to check.

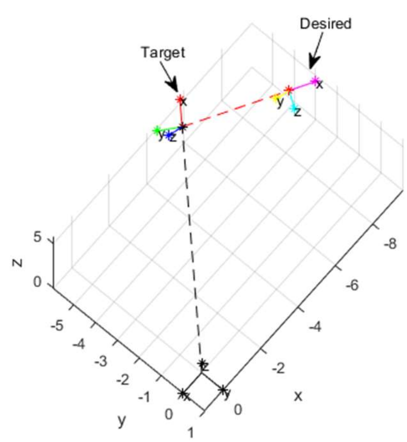

Fig 3.7: Position of the reference frame D and T at random time

Since these 2 x axis are pointing to the origin of another reference frame and the radius relationship is generally correct, we may say they all run in the designed trajectories. This result is driven from “two_reference_frame.m”.

4 SATELLITE MOTION

Refer to the satellite control paper[1], the purpose is to control a chaser satellite move to the desired reference frame, which is designed to rotate around a target satellite and assume the target satellite is in the Molniya orbit. To prove the stability of the system, the author designed a Lyapunov function V, and evidence the $\dot{V} \leq 0, V \geq 0$.

Chapter 3.1 and 3.2 indicate or prove the attitude representation equation and the rotational kinematic equation, even most of the equations are proved in [5], [6], [8], there are still some equations deserved to be mentioned.

1) **Combination of Transformation**

$$\begin{matrix} r^T = q_{T/I}^* r^I q_{T/I} \\ r^D = q_{D/T}^* r^T q_{D/T} \end{matrix} \rightarrow r^D = q_{D/T}^* q_{T/I}^* r^I q_{T/I} q_{D/T} \rightarrow q_{D/I} = q_{T/I} q_{D/T}$$

where r^T denotes the r seen in the T reference frame, $q_{T/I}$ is reference frame T changed from reference frame I. Likewise, dual quaternion also obeys this rule, e.g., $\hat{q}_{D/T} = \hat{q}_{T/I}\hat{q}_{D/T}$. This rule is also used in 3.3.

2) **Inverse of Transformation**

From the 1) the by times $q_{T/I}^*$ both sides we get $q_{D/T} = q_{T/I}^* q_{D/I}$, since it is a unit quaternion. Another way is just using the footnote, $q_{D/T} = q_{I/T} q_{D/I}$. So $q_{I/T} = q_{T/I}^*$.

3) **Angular Velocity in Dual Quaternion Form**

Even a vector $\vec{w}_{B/D}^B$ can be rotated in dual quaternion form, but since the vector has no information of the displacement $\vec{r}_{B/D}^B$ or $\vec{r}_{B/D}^D$ respect to the reference frame D, it's impossible to rotate it which is denoted in another reference frame B. This requires the vector needs to connect to the $\vec{r}_{B/D}^B$ or $\vec{r}_{B/D}^D$, or denotes as a modified dual velocity[10]:

$$\widehat{w}_{Y/X}^X = w_{Y/X}^X + \epsilon\left(v_{Y/X}^X + w_{Y/X}^X \times r_{X/Y}^X\right), v_{Y/X}^X = \left(\vec{v}_{Y/X}^X, 0\right)$$

where $\vec{v}_{Y/X}^X$ denotes the velocity vector of reference frame Y respect to X and seen in the X.

This definition also uses the line vector representation method in the Plücker coordinate, where the line is usually defined as $(\vec{l}, \vec{p} \times \vec{l})$. The line has a dual quaternion representation:

$$\hat{l} = l + \epsilon p \times l$$

with $\hat{l}$ is the dual quaternion for the line, p is the pure quaternion of a point p on the line l, that $p = (\vec{p}, 0)$, and l is a pure quaternion of the line direction. Here we denote the line vector use $\left(\vec{w}_{Y/X}^X, \vec{v}_{Y/X}^X + \vec{r}_{Y/X}^X \times \vec{w}_{Y/X}^X\right)$, so the dual quaternion is $\hat{l}_w = w_{Y/X}^X + \epsilon(v_{Y/X}^X + r_{Y/X}^X \times w_{Y/X}^X)$. With the proof from [8], we certain the $\hat{l}_w$ is able to transform with $\hat{q}_{Y/X}$, as a result.

4) **Derivative of Angular Velocity in Dual Quaternion Form**

Driven from 3) the derivative of angular velocity in dual quaternion from is represented as

$$\begin{aligned}\dot{\hat{w}}_{Y/X}^{X} &= \dot{w}_{Y/X}^{X} + \epsilon(\dot{v}_{Y/X}^{X} - \dot{w}_{Y/X}^{X} \times r_{Y/X}^{X} - w_{Y/X}^{X} \times \dot{r}_{Y/X}^{X}) \\ &= \alpha_{Y/X}^{X} + \epsilon(a_{Y/X}^{X} - \alpha_{Y/X}^{X} \times r_{Y/X}^{X} - w_{Y/X}^{X} \times a_{Y/X}^{X})\end{aligned}$$

where is same as the $\hat{\alpha}_{Y/X}^{X}$ in the [1].

4.1 MOLYNIA ORBIT

Satellite motion around earth is an ellipse orbit if we stand in the earth-centered inertial frame (ECI), which is also called earth-centered equatorial system. In order to position the satellite, we require 6 parameter, namely classical orbital elements (COEs), including 5 for the orbit (Tab 4.1) and 1 true anomaly υ [11].

COE	Denotation	Value
Semimajor axis	a	2.66×10^7 m
Eccentricity	e	0.74
Inclination	i	63.4°
RAAN	Ω	329.6°
Argument of perigee	ω	270°

Tab 4.1: Classical orbital elements of Molniya orbit

Any elliptical orbit can be transformed from the ECI frame with a rotation around $\hat{K}$ with Ω degree, and a sequent one around the new $\hat{I}$, $\hat{I}'$, with inclination i and a final rotation around the new $\hat{K}$, $\hat{K}'$, with ω [12]. This progress can be described in the Euler rotation matrix as $Rzxz(\Omega, i, \omega)$, so that also as a quaternion.

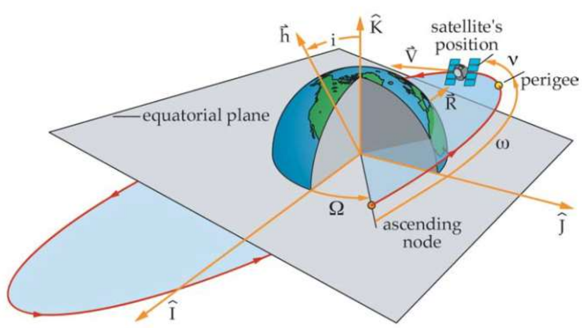

Fig 4.1: Reference frame transformation[11]

After the transformation, we can get the state vector in Molniya orbit represented in the cartesian coordinate with the parameters from Tab 4.1. Since the I axis of the satellite reference frame is in the same direction of displacement of the satellite according to ECI, $\vec{r}^{I}_{T/I}$, such that the angular velocity of it decides the true anomaly.

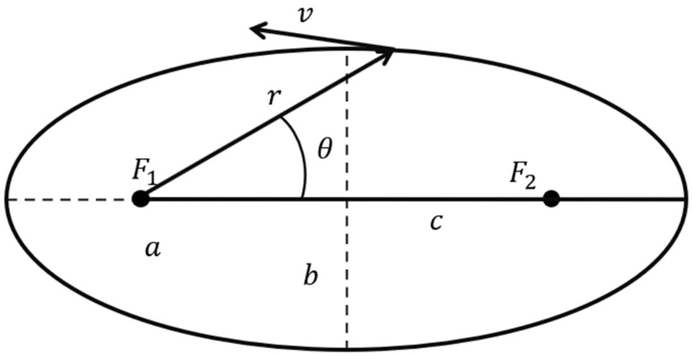

Fig 4.2: Satellite orbit parameter in 2D plane

In addition, the specific angular momentum h required for the algorithm is calculated by formula $h = \vec{r} \times \vec{v}$ [12], which are denoted as Fig 4.2. The simplest way to get h is to use the velocity and radius vector at perigee or apogee[12], when the true anomaly is 0 or π.

Rotating in a period in script "Molniya_orbit.m", the reference frame and trajectory are generated here for reference. Note that here the orbit is not generated by the actual time but more like a scan of true anomaly from 0 to 2π.

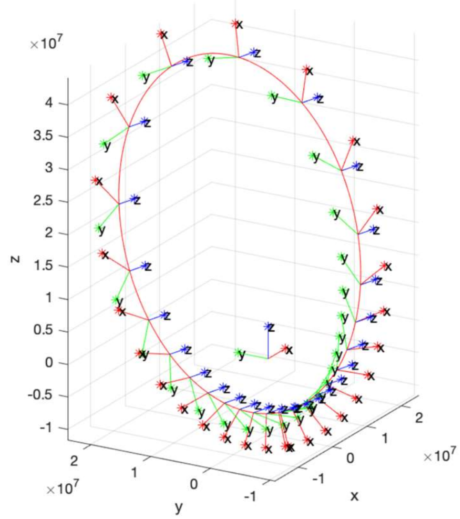

Fig 4.3: Molniya orbit

4.2 DESIRED REFERENCE FRAME

Based on the circular motion we have done in chapter 3.2, we can set up the first 2 phases of the reference frame D [1] respect to the reference frame T, the satellite reference frame, yet seen in the reference frame T. This result is from “two_phase.m”, where the angular velocity in phase 2 is smaller for better demonstration and faster calculation speed.

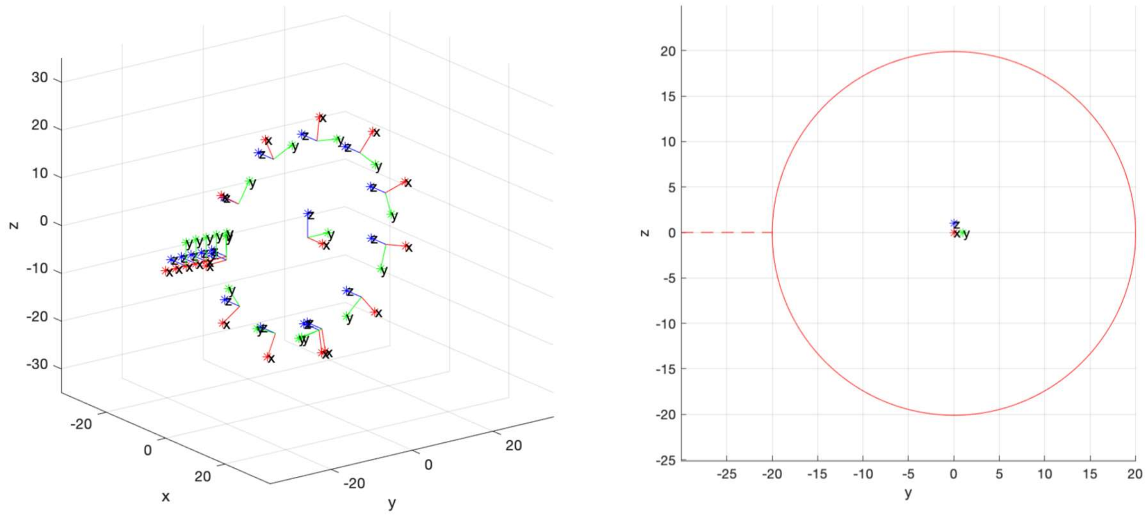

Fig 4.4: Desired reference frame motion respect to reference frame T

Since the angular velocity also infects the cumulation, for the designed angular velocity in [1], timestep should be set up to 1e-4 or a smaller level to simulate the designed trajectory, nevertheless, the running time ascends unacceptably.

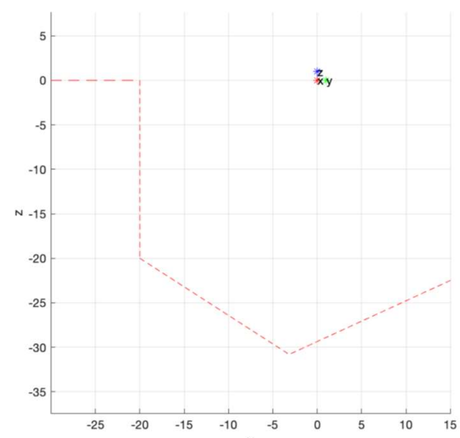

Fig 4.5: Circular motion with angular velocity in 1e3 level but timestep in 1e-3 level

A normal timestep will leads to the chaser satellite problem (Fig 4.5), that it will be always chasing the desired reference frame without reaching it. Thus, this report only simulates the phase 1 process for greater accuracy.

4.3 RESULT REPRODUCE

The result in [1] all simulated in the time space, while the Molniya orbit in 4.1 is in the true anomaly space. Thus, according the [12], the trajectory and velocity can be integrated through acceleration, if we want to draw the trajectory in time space.

$$\vec{a}^{I}_{T/I} = -\frac{\mu \vec{r}^{I}_{T/I}}{\left(\vec{r}^{I}_{T/I}\right)^{3}} + \vec{p}$$

where the μ is standard gravitational parameter of earth, $\mu = 3.986 \times 10^{14}\mathrm{m}^{3}\mathrm{s}^{-2}$, and $\vec{p}$ is related to derivative of Ω and ω [12], yet for Molniya orbit they will barely change. To get the $\hat{q}_{T/I}$, we also need

$$\vec{w}^{I}_{T/I} = \frac{\vec{r}^{I}_{T/I} \times \vec{v}^{I}_{T/I}}{\left(\vec{r}^{I}_{T/I}\right)^{2}}$$

Using the alike idea from 3.3, we can draw the trajectory of Molniya orbit and the desired reference frame. Note that from the period of Molniya orbit [12] is calculated as: $T = 2\pi a^{\frac{3}{2}}/\sqrt{\mu} = 4.3175 \times 10^{4}\ \mathrm{s} \approx 720\mathrm{min}$. Given the level of the cumulation goal, the timestep yields to 1s, such that the simulation of phase 2 is impossible.

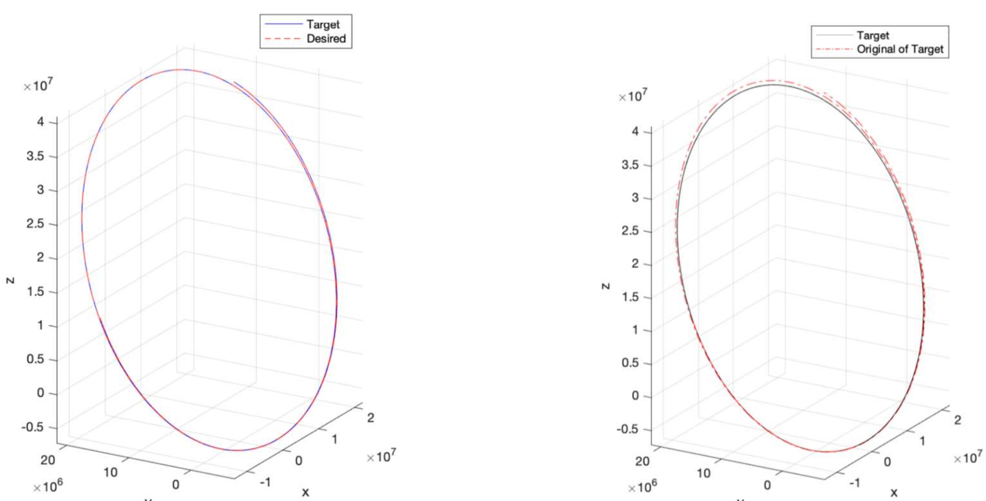

Fig 4.6: Target and desired reference frame. 1) left: target and desired 2) right: target and Molniya

This huge timestep causes also large deviation in trajectory of Molniya orbit (Fig4.6), they can't neither fit perfectly to their own trajectory(left) after a period, nor match exactly the designed one(right).

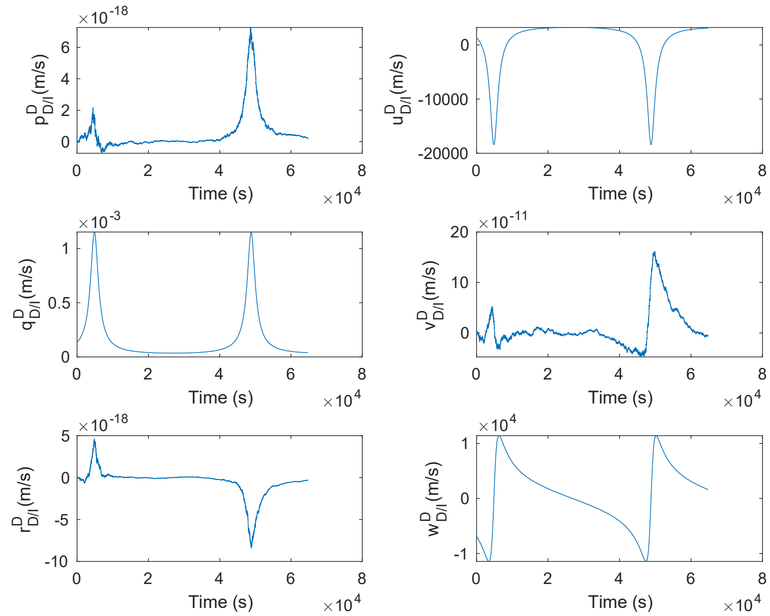

Fig 4.7: $\vec{w}_{D/I}^{D}$ and $\vec{v}_{D/I}^{D}$

When see the $\vec{w}_{D/I}^{D} = \left[p_{D/I}^{D}, q_{D/I}^{D}, r_{D/I}^{D}\right]^{T}$ and $\vec{v}_{D/I}^{D} = \left[u_{D/I}^{D}, v_{D/I}^{D}, w_{D/I}^{D}\right]^{T}$ in 1.5 period, they contain part of the periodical signals but differ from [1]. This may be triggered by the large timestep, or the calculation method of angular velocity $\vec{w}_{D/I}^{D}$, that the author may also get the angular velocity from angular acceleration while I used the cross product of radius and velocity vector. In addition, we only use the phase 1 for simulation, which lasts only 400s, and set phase 2's angular velocity $\vec{w}_{D/I}^{D}$ and linear velocity $\vec{v}_{D/I}^{D}$ to 0, while they should not be, thus, the 2 velocity we have for describing reference frame D are more like belong to reference frame T, when comparing 400s to the whole simulation time.

Considering the chaser reference frame B, a nonlinear system which is represented in dual quaternion and needs to integrate from acceleration, the timestep plays an important role for a more than 700 mins simulation. In order to ensure the precision of the simulation, powerful computation ability is essential.

5 CONLUSION

This report concludes the quaternion and dual quaternion utilization in the rigid body motion by visualizing their rotation and transformation properties. Limited by the aerospace knowledge of the satellite, only the orbit and kinetic of the satellite is represented here, and the dynamic equation of the quaternion and dual quaternion is not included in this report. Even succeed in the reshow of the desired reference frame motion, yet due to the bounded computation ability, the motion phases are redesigned, consequently the result is unable to compared with the research paper for correctness checking. An alternative solution of the integration problem is to linearize the known model of the reference frame T, D and B or use Koopman operator theory, which are unproved for certainty but still have probability.

In general, we do see the advantages of quaternion and dual quaternion: when computing reference frame transformation, they avoid multiple matrix operation and show up compactness; when rotating vectors among different reference frame, they can update the vector only in one equation, especially for angular velocity. One thing we need to take care of in rotation and transformation, is the quaternion and dual quaternion should be multiplied at both sides of the vector, but with one of the sides in conjugate form, which can be confusing when dealing with transforming vector between reference frames forwards and backwards. When coding in Matlab, applying them for multiple transformations requires recursively creating the classes. As a result, if the trajectory is generated from the accumulation, the simulation speed is restricted. Perhaps an application in Python or other language will accelerate the computation compared to homogeneous matrix multiplication in the same language.

In summary, the quaternion and dual quaternion are an alternative way to describe the rigid body motion: they obey the Euler's theorem for rigid body displacement. The two have self-consistent form in the basic operations, attitude representation, and kinetic equations, thus, in the cumulation computation, e.g. updating in dual quaternion form, they demonstrate compactness and less computation if the program is well designed.

Reference:

[1] N. Filipe and P. Tsiotras, "Adaptive position and attitude-tracking controller for satellite proximity operations using dual quaternions," *J. Guid. Control. Dyn.*, vol. 38, no. 4, pp. 566–577, 2015.

[2] W. R. Hamilton, " XLII. On quaternions; or on a new system of imaginaries in algebra ," *London, Edinburgh, Dublin Philos. Mag. J. Sci.*, vol. 34, no. 229, pp. 294–297, 1849.

[3] D. Hoag, "Apollo Guidance and Navigation Considerations of Apollo IMU Gimbal Lock," *Canbridge MIT Instrum. Lab.*, pp. 1–12, 1963.

[4] MathsPoetry, "Gimbal lock: two out of the three gimbals are in the same plane, one degree of freedom is lost," 2020. https://commons.wikimedia.org/w/index.php?curid=5949077.

[5] B. Graf, "Quaternions and dynamics." 2008, [Online]. Available: http://arxiv.org/abs/0811.2889.

[6] H. Parwana and M. Kothari, "Quaternions and attitude representation," *arXiv*. 2017.

[7] C. Clifford, *Preliminary sketch of biquaternions*, vol. s1-4, no. 1. 1871.

[8] Y. Jia, "Dual Quaternions," *Handouts Cl. CS577 Univ. Iowa*, 2019.

[9] G. Chirikjian, A. Kyatkin, and A. Buckingham, "Erratum: Engineering Applications of Noncummutative Harmonic Analysis: With Emphasis on Rotation and Motion Groups," *Appl. Mech. Rev.*, vol. 55, no. 1, pp. B20–B20, 2002.

[10] I. S. Fischer, "A complete notation for dual velocity," *J. Mech. Des. Trans. ASME*, vol. 114, no. 2, pp. 263–268, 1992.

[11] J. Jon Sellers, W. J. Astore, R. B. Giffen, and W. J. Larson, *Understanding Space - An Introduction to Astronautics*. Primis, 2004.

[12] H. D. Curtis, *Orbital mechanics for engineering students*. Butterworth-Heinemann, 2013.