

Article

Not peer-reviewed version

Linearized Expressions of 3D Rotational Motion

[Yinlong Liu](#) *

Posted Date: 17 February 2025

doi: 10.20944/preprints202405.0982.v2

Keywords: Rotation Motion; $SO(3)$; Clifford Torus; Unit Quaternion Sphere



Preprints.org is a free multidisciplinary platform providing preprint service that is dedicated to making early versions of research outputs permanently available and citable. Preprints posted at Preprints.org appear in Web of Science, Crossref, Google Scholar, Scilit, Europe PMC.

Copyright: This open access article is published under a Creative Commons CC BY 4.0 license, which permit the free download, distribution, and reuse, provided that the author and preprint are cited in any reuse.

Article

Linearized Expressions of 3D Rotational Motion

Yinlong Liu 

Technical University of Munich; yinlong.liu@tum.de

Abstract: Rotation motion in a three-dimensional physical world refers to an angular displacement of an object around a specific axis in $\mathbb{R}^3$. It is typically formulated as a non-linear and non-convex motion due to the nonlinearity and nonconvexity of $\mathbb{SO}(3)$. However, this paper proposes a new perspective that the 3D rotation motion can be expressed by a linear system without dropping any constraints and increasing any singularities. Moreover, two frequent cases, i.e., $\angle(\mathbf{R}\mathbf{x}, \mathbf{y}) = 0$ and $\angle(\mathbf{R}\mathbf{x}, \mathbf{y}) = \frac{\pi}{2}$, in computer vision and robotics that can be expressed linearly are deeply discussed in this paper.

Keywords: rotation motion; $\mathbb{SO}(3)$; Clifford Torus; unit quaternion sphere

1. Introduction

Rotational motion [1] in the three-dimensional physical world is regarding to the circular movement of an object around a fixed axis [2]. It is a basic concept and commonly used in physics, computer graphics, computer vision, robotics, aerospace, and other scientific fields [3–6]. More specifically, in many practical applications, rotational motion is not only an important way of constructing the motion of the target system [7–9] but also an important state parameter to be estimated [10–12]. However, rotation motion is typically formulated as a non-linear and non-convex process [11–13]. It not only makes some linear algorithms, such as Kalman filter [14,15], impossible to directly apply [16], but also makes solving the related problem more sophisticated and difficult [13,17].

In this paper, a new perspective is proposed that **the rotation motion can be linearized without dropping any constraints and increasing any singularities**. It is particularly worth pointing out that there are no special and advanced techniques, such as Lie groups and Lie algebras [18,19], for the methods used in this paper. Only some basic knowledge of linear algebra [20–22] is required to understand the ideas presented in this paper.

The structure of this paper is as follows: it first reviews rotational motion in the second section, then presents the linearized expression for rotation in the third section. The fourth and fifth sections discuss two special cases, i.e., $\tau = 0$ and $\tau = \frac{1}{2}\pi$, providing in-depth analysis.

2. Rotation Motion

Mathematically, given a point $\mathbf{x} \in \mathbb{R}^3$, if it is rotated about a fixed unit vector $\mathbf{r} \in \mathbb{S}^2$ through the origin and the rotated angle is θ , then the rotation motion can be formulated by multiplication of a matrix and a vector,

$$\mathbf{R}\mathbf{x} \rightarrow \mathbf{y} \quad (1)$$

where $\mathbf{y} \in \mathbb{R}^3$ is the rotated point by $\mathbf{x}$ and $\mathbf{R}$ is a 3×3 rotation matrix. There are some properties regarding the rotation motion [23],

- Shape and Size: Rotation does not alter the shape or size of the object. More Specifically,
 1. $\|\mathbf{R}\mathbf{x}\| = \|\mathbf{R}\| \cdot \|\mathbf{x}\| = \|\mathbf{x}\|$. It means the length of the point $\mathbf{x}$ will not be changed by rotation motion. Without loss of generality, the rotation motion in this paper is studied with unit norm vectors, i.e., $\mathbf{x} \in \mathbb{S}^2$ on the unit sphere surface, which represents a unit direction vector in the 3D physical world.
 2. $\angle(\mathbf{R}\mathbf{x}_1, \mathbf{R}\mathbf{x}_2) = \angle(\mathbf{x}_1, \mathbf{x}_2)$ where $\mathbf{x}_1, \mathbf{x}_2 \in \mathbb{S}^2$ are any two unit vectors. It means the angle (structure) of the object is unchanged.

- **Axis of Rotation:** Rotational motion occurs around a fixed axis. Given a vector $x \in \mathbb{S}^2$, if $\mathbf{R}x = x$, it means the after the rotation $\mathbf{R}$, the vector direction x remains unchanged, and the vector x must be parallel to the rotation axis r , which is from Euler's rotation theorem and screw theory [24–27]. In addition,

$$\mathbf{R}x = x \Rightarrow (\mathbf{R} - \mathbf{I}_{3 \times 3})x = 0 \quad (2)$$

where $\mathbf{I}_{3 \times 3}$ is the 3×3 identity matrix. Algebraically, Eq.(2) means the rotation axis direction r lies in the null space of $\mathbf{R} - \mathbf{I}_{3 \times 3}$. Alternatively, let $\mathbf{R}x = \lambda x$, then the rotation axis direction r is an eigenvector of $\mathbf{R}$ corresponding to the eigenvalue $\lambda = 1$.

At first glance, the rotation motion is a linear process, which might be misled by the matrix representation Eq.(1). Unfortunately, $\mathbf{R}$ is not a regular matrix in $\mathbb{R}^{3 \times 3}$. A rotation only has three degrees of freedom, since it can be determined by a rotation axis $r \in \mathbb{S}^2$ and a rotation angle $\theta \in [0, \pi]$. Formally, $\mathbf{R} \in \mathbb{SO}(3)$, and a rotation can be represented by a matrix and non-linear constraints,

$$\mathbf{R} = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix}, \quad \text{s.t.} \quad \mathbf{R}^T \mathbf{R} = \mathbf{I}_{3 \times 3}, \det(\mathbf{R}) = +1 \quad (3)$$

According to the famous Rodrigues' rotation formula [28,29], given the rotation axis $r = [r_1 \ r_2 \ r_3]^T \in \mathbb{S}^2$ and rotation angle θ , the matrix can be constructed by

$$\mathbf{R} = \exp(\theta[r]_{\times}) = \mathbf{I}_{3 \times 3} + \sin(\theta)[r]_{\times} + (1 - \cos(\theta))[r]_{\times}^2, \quad [r]_{\times} = \begin{bmatrix} 0 & -r_3 & r_2 \\ r_3 & 0 & -r_1 \\ -r_2 & r_1 & 0 \end{bmatrix} \quad (4)$$

Moreover, a rotation matrix can be represented by a quaternion vector with unit length [30,31], such that $q = [q_0 \ q_1 \ q_2 \ q_3]^T \in \mathbb{S}^3$. Specifically,

$$\mathbf{R} = \begin{bmatrix} q_0^2 + q_1^2 - q_2^2 - q_3^2 & 2(q_1q_2 - q_0q_3) & 2(q_1q_3 + q_0q_2) \\ 2(q_1q_2 + q_0q_3) & q_0^2 - q_1^2 + q_2^2 - q_3^2 & 2(q_2q_3 - q_0q_1) \\ 2(q_1q_3 - q_0q_2) & 2(q_2q_3 + q_0q_1) & q_0^2 - q_1^2 - q_2^2 + q_3^2 \end{bmatrix} \quad (5)$$

It is worth noting that many different forms of parameter representation can represent the same rotation, and each form of representation has its appropriate application scenario [32–35]. For example, since a rotation can be determined by a fixed rotation axis $r \in \mathbb{S}^2$ and a rotation angle $\theta \in [0, \pi]$, then the rotation can be represented by axis-angle representation $\theta \cdot r$ [36], which geometrically is a point in a 3D ball of radius π [29] and has been widely used in pose estimation problems [29,37,38]. Nonetheless, throughout most of this paper, quaternions are used to represent rotations [30,31] and thus to illustrate the proposed linearization theory.

“Anyone who has ever used any other parametrization of the rotation group will, within hours of taking up the quaternion parametrization, lament his or her misspent youth” [33,39].

3. Linear Expressions for Rotation Motion

Given $a = [a_1 \ a_2 \ a_3]^T \in \mathbb{S}^2$ and a rotation $\mathbf{R} \in \mathbb{SO}(3)$, the rotation motion can be formulated as

$$\mathbf{R}a = b \Leftrightarrow \angle(\mathbf{R}a, b) = 0 \Leftrightarrow b^T \mathbf{R}a = 1 \quad (6)$$

where $\mathbf{b} = \begin{bmatrix} b_1 & b_2 & b_3 \end{bmatrix} \in \mathbb{S}^2$ is rotated by $\mathbf{a}$. This formulation is actually a special case of the general rotational motion. Generally, given $\mathbf{a}, \mathbf{b} \in \mathbb{S}^2$ and $\mathbf{R} \in \mathbb{SO}(3)$ whose quaternion expression is $\mathbf{q} \in \mathbb{S}^3$, consider an arbitrary angle $\tau \in [-\pi, \pi]$, there is

$$\angle(\mathbf{R}\mathbf{a}, \mathbf{b}) = \tau \Leftrightarrow \mathbf{b}^T \mathbf{R}\mathbf{a} = \cos(\tau) \Leftrightarrow \mathbf{q}^T \mathbf{M}\mathbf{q} = \cos(\tau) \quad (7)$$

Here $\mathbf{M}$ is constructed by $\mathbf{a}$ and $\mathbf{b}$ [30,40]. Specifically,

$$\mathbf{M} = \begin{bmatrix} 0 & -a_1 & -a_2 & -a_3 \\ a_1 & 0 & a_3 & -a_2 \\ a_2 & -a_3 & 0 & a_1 \\ a_3 & a_2 & -a_1 & 0 \end{bmatrix}^T \begin{bmatrix} 0 & -b_1 & -b_2 & -b_3 \\ b_1 & 0 & -b_3 & b_2 \\ b_2 & b_3 & 0 & -b_1 \\ b_3 & -b_2 & b_1 & 0 \end{bmatrix} \quad (8)$$

When $\tau = 0$, Eq.(7) degenerates back to Eq.(6).

It is worth noting that there are two facts concerning $\angle(\mathbf{R}\mathbf{a}, \mathbf{b}) = \tau$,

1. Not all rotations in $\mathbb{SO}(3)$ can satisfy the given rotation motion.
2. There are an infinite number of rotations that satisfy the given rotation motion.

For the first point, it is understandable. For the second point, observe $\angle(\mathbf{R}\mathbf{a}, \mathbf{b}) = \tau \Rightarrow \angle(\mathbf{R}'(\theta)\mathbf{R}\mathbf{a}, \mathbf{b}) = \tau$, where $\mathbf{R}'(\theta)$ is a rotation whose rotation axis is $\mathbf{R}\mathbf{a}$ and rotation angle is θ ; θ is an arbitrary angle. It means if $\mathbf{R}$ satisfy $\angle(\mathbf{R}\mathbf{a}, \mathbf{b}) = \tau$, then there are an infinite number of rotations $\mathbf{R}'(\theta)\mathbf{R}$ that also satisfy the given rotation motion. Therefore, given a rotation constraint $\angle(\mathbf{R}\mathbf{a}, \mathbf{b}) = \tau$, all rotations that satisfy the given rotation motion should be in some special regions in $\mathbb{SO}(3)$.

By conducting eigendecomposition of the real symmetric matrix $\mathbf{M} = \mathbf{U}\mathbf{\Lambda}\mathbf{U}^T$ [22,41], where and $\mathbf{U} = \begin{bmatrix} u_0 & u_1 & u_2 & u_3 \end{bmatrix} \in \mathbb{R}^{4 \times 4}$ is an orthogonal matrix $\mathbf{U}\mathbf{U}^T = \mathbf{I}_{4 \times 4}$. Observe

$$\det(\mathbf{M} - \lambda \mathbf{I}_{4 \times 4}) = 0 \Rightarrow (\lambda^2 - 1)^2 = 0 \quad (9)$$

Therefore, the eigenvalue matrix of $\mathbf{M}$ is

$$\mathbf{\Lambda} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & -1 \end{bmatrix} \quad (10)$$

Accordingly, $\mathbf{q}^T \mathbf{U}\mathbf{\Lambda}\mathbf{U}^T \mathbf{q} = \cos(\tau)$.

Let $\mathbf{p} = [p_0, p_1, p_2, p_3]^T = \mathbf{U}^T \mathbf{q}$, then $\mathbf{q} = \mathbf{U}\mathbf{p} = p_0 \mathbf{u}_0 + p_1 \mathbf{u}_1 + p_2 \mathbf{u}_2 + p_3 \mathbf{u}_3$. Observe

$$\begin{cases} \mathbf{p}^T \mathbf{p} = \mathbf{q}^T \mathbf{U}\mathbf{U}^T \mathbf{q} = p_0^2 + p_1^2 + p_2^2 + p_3^2 = 1 \\ \mathbf{q}^T \mathbf{U}\mathbf{\Lambda}\mathbf{U}^T \mathbf{q} = \mathbf{p}^T \mathbf{\Lambda} \mathbf{p} = p_0^2 + p_1^2 - p_2^2 - p_3^2 = \cos(\tau) \end{cases} \Rightarrow \begin{cases} p_0^2 + p_1^2 = \frac{1}{2}(1 + \cos(\tau)) \\ p_2^2 + p_3^2 = \frac{1}{2}(1 - \cos(\tau)) \end{cases} \quad (11)$$

Therefore, it can be formulated

$$\begin{cases} p_0 = \frac{\sqrt{2(1 + \cos(\tau))}}{2} \cos(\phi), & p_1 = \frac{\sqrt{2(1 + \cos(\tau))}}{2} \sin(\phi) \\ p_2 = \frac{\sqrt{2(1 - \cos(\tau))}}{2} \cos(\psi), & p_3 = \frac{\sqrt{2(1 - \cos(\tau))}}{2} \sin(\psi) \end{cases} \quad (12)$$

As a consequence, the possible rotation q can be reformulated linearly

$$q = p_0 u_0 + p_1 u_1 + p_2 u_2 + p_3 u_3 \quad (13)$$

$$= \frac{\sqrt{2(1+\cos(\tau))}}{2} (\cos(\phi) u_0 + \sin(\phi) u_1) + \frac{\sqrt{2(1-\cos(\tau))}}{2} (\cos(\psi) u_2 + \sin(\psi) u_3) \quad (14)$$

where $\phi, \psi \in [-\pi, \pi]$. This result clearly shows that the rotational motion can be expressed in linear forms.

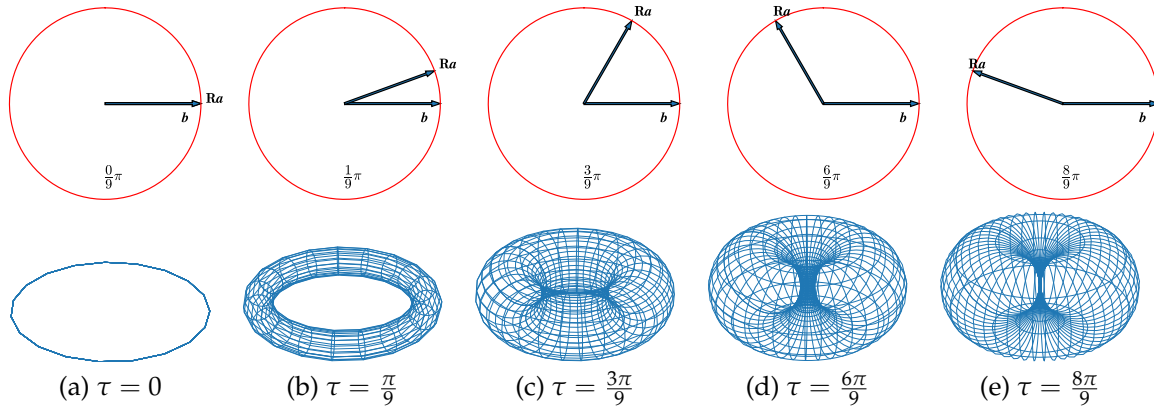


Figure 1. The illustration of genera cases of rotation constraints and their corresponding tori, which are consisting of all rotations that satisfy their respective constraint, in $\mathbb{R}^3$ via stereographic projection.

Geometrically, if $\angle(\mathbf{Ra}, \mathbf{b}) = \tau$, then all possible rotation q that satisfy the constraint will be in a two-dimensional torus in $\mathbb{S}^3$. For better understanding, some special cases can be visualized, i.e., $\tau = \{0, \frac{\pi}{9}, \frac{3\pi}{9}, \frac{6\pi}{9}, \frac{8\pi}{9}\}$ via stereographic projection [42,43], as in Fig 1.

It is worth noting that when $\tau = \{0, \pi\}$, i.e., $\mathbf{Ra} = \pm \mathbf{b}$, the torus in $\mathbb{S}^3$ is degenerated into a one-dimensional circle, more specifically, a great circle [44–46] in $\mathbb{S}^3$. Formally,

$$\begin{aligned} (a) : \quad & \text{when } \tau = 0 \Rightarrow q = \cos(\phi) u_0 + \sin(\phi) u_1 \\ (b) : \quad & \text{when } \tau = \pi \Rightarrow q = \cos(\psi) u_2 + \sin(\psi) u_3 \end{aligned} \quad (15)$$

In addition, when $\tau = \frac{\pi}{2}$, the torus is becoming to the Clifford torus in $\mathbb{S}^3$ [47,48], since it has the equal *diameters* in cross directions. Specifically,

$$q = \frac{\sqrt{2}}{2} (\cos(\phi) u_0 + \sin(\phi) u_1) + \frac{\sqrt{2}}{2} (\cos(\psi) u_2 + \sin(\psi) u_3) \quad (16)$$

4. Special Case I: Great Circle in $\mathbb{S}^3$

If $\tau = 0$, which means $\mathbf{Ra} = \mathbf{b}$, it is one of the most common cases in practical applications [33,37]. Then according to Eq. (15) all rotations that satisfy $\mathbf{Ra} = \mathbf{b}$ should be in

$$q = \cos(\phi) u_0 + \sin(\phi) u_1 \quad (17)$$

where u_0 and u_1 are the basis vectors in $\mathbb{S}^3$, and they depend on $\mathbf{M}$, i.e., $\mathbf{a}$ and $\mathbf{b}$. Geometrically speaking, q is a unit vector in $\mathbb{R}^4$, which is a linear combination of the two vectors u_0 and u_1 . In addition, $q^T q = 1$, which means q is on the surface of the unit quaternion sphere. Therefore, the region where all possible rotations satisfy a specific rotation motion is a great circle in $\mathbb{S}^3$ [44].

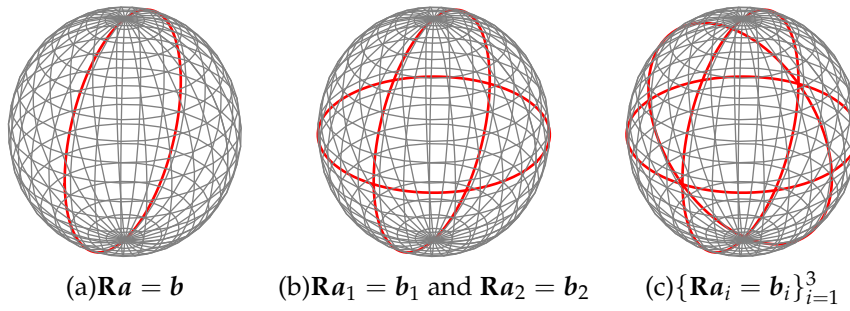


Figure 2. The illustration of great circles in $\mathbb{S}^3$; for better visualization, they are imitatively drawn in $\mathbb{R}^3$. (a) Given $\mathbf{R}\mathbf{a} = \mathbf{b}$, all possible solutions of the to-be-solved rotation will be in a circle-shape region on $\mathbb{S}^3$. (b) Given two rotation motions, there are two circles on the surface, and the intersections of the circles are the solutions that meet both rotation motions. Note that the intersections are symmetrical on the surface, and they are corresponding to the same rotation matrix. (c) Given more rotation motions, there are more circles. The intersections of the circles should be the rotation solution that meets all given rotation motions.

In order to determine a unique rotation, two independent constraints are required. Specifically, given two pairs of inputs $\mathbf{R}\mathbf{a}_1 = \mathbf{b}_1$ and $\mathbf{R}\mathbf{a}_2 = \mathbf{b}_2$, two circles can be calculated according to Eq.(17), which is however not easy.

$$\begin{cases} \mathbf{q} = \cos(\phi_1)\mathbf{u}_{1,0} + \sin(\phi_1)\mathbf{u}_{1,1} \\ \mathbf{q} = \cos(\phi_2)\mathbf{u}_{2,0} + \sin(\phi_2)\mathbf{u}_{2,1} \end{cases} \Rightarrow ? \quad (18)$$

In fact, it does not need to be that complicated. Since $\mathbf{U}^T\mathbf{U} = \mathbf{I}_{4 \times 4}$, there will be

$$\begin{cases} (\cos(\phi)\mathbf{u}_0^T + \sin(\phi)\mathbf{u}_1^T)\mathbf{u}_2 = 0 \\ (\cos(\phi)\mathbf{u}_0^T + \sin(\phi)\mathbf{u}_1^T)\mathbf{u}_3 = 0 \end{cases} \Rightarrow \begin{cases} \mathbf{q}^T\mathbf{u}_2 = 0 \\ \mathbf{q}^T\mathbf{u}_3 = 0 \end{cases} \quad (19)$$

Consequently, given $\mathbf{R}\mathbf{a} = \mathbf{b}$, there will be $[\mathbf{u}_2^T, \mathbf{u}_3^T]\mathbf{q} = 0$, where $\mathbf{u}_2$ and $\mathbf{u}_3$ are constructed by $\mathbf{a}$ and $\mathbf{b}$. Given $\{\mathbf{a}_1, \mathbf{b}_1\}$ and $\{\mathbf{a}_2, \mathbf{b}_2\}$, there will be

$$\begin{cases} [\mathbf{u}_{1,2}^T & \mathbf{u}_{1,3}^T]\mathbf{q} = 0 \\ [\mathbf{u}_{2,2}^T & \mathbf{u}_{2,3}^T]\mathbf{q} = 0 \end{cases} \Rightarrow [\mathbf{u}_{1,2}^T & \mathbf{u}_{1,3}^T & \mathbf{u}_{2,2}^T & \mathbf{u}_{2,3}^T]\mathbf{q} = 0 \stackrel{def}{\iff} \mathbf{K}_{4 \times 4}\mathbf{q} = 0 \quad (20)$$

Similarly, given N inputs, $\{\mathbf{a}_i, \mathbf{b}_i\}_{i=1}^N$, there will be

$$\begin{cases} \mathbf{R}\mathbf{a}_1 = \mathbf{b}_1 \\ \mathbf{R}\mathbf{a}_2 = \mathbf{b}_2 \\ \dots \\ \mathbf{R}\mathbf{a}_N = \mathbf{b}_N \end{cases} \Rightarrow \begin{cases} [\mathbf{u}_{1,2}^T & \mathbf{u}_{1,3}^T]\mathbf{q} = 0 \\ [\mathbf{u}_{2,2}^T & \mathbf{u}_{2,3}^T]\mathbf{q} = 0 \\ \dots \\ [\mathbf{u}_{N,2}^T & \mathbf{u}_{N,3}^T]\mathbf{q} = 0 \end{cases} \stackrel{def}{\iff} \mathbf{K}_{2N \times 4}\mathbf{q} = 0 \quad (21)$$

Up until this point, the rotation motion can be surprisingly linearly solved.

4.1. Eigenvector of $\mathbf{M}$

It is worth noting that the linear expression of rotation motion is closely related to $\mathbf{U} = [\mathbf{u}_0 \ \mathbf{u}_1 \ \mathbf{u}_2 \ \mathbf{u}_3]$. Due to $\mathbf{M}\mathbf{U} = \mathbf{U}\mathbf{\Lambda}$, there will be

$$\mathbf{M}\mathbf{u}_0 = 1 \cdot \mathbf{u}_0; \quad \mathbf{M}\mathbf{u}_1 = 1 \cdot \mathbf{u}_1; \quad \mathbf{M}\mathbf{u}_2 = -1 \cdot \mathbf{u}_2; \quad \mathbf{M}\mathbf{u}_3 = -1 \cdot \mathbf{u}_3 \quad (22)$$

It is easy to confirm [49,50],

$$\mathbf{M} \begin{bmatrix} 0 & -(a-b)^T \\ -(a-b) & [a+b]_{\times}^T \end{bmatrix} = -1 \cdot \begin{bmatrix} 0 & -(a-b)^T \\ -(a-b) & [a+b]_{\times}^T \end{bmatrix} \stackrel{\text{def}}{=} -1 \cdot \Omega^- \quad (23)$$

$$\mathbf{M} \begin{bmatrix} 0 & -(a+b)^T \\ -(a+b) & [a-b]_{\times}^T \end{bmatrix} = +1 \cdot \begin{bmatrix} 0 & -(a+b)^T \\ -(a+b) & [a-b]_{\times}^T \end{bmatrix} \stackrel{\text{def}}{=} +1 \cdot \Omega^+ \quad (24)$$

Therefore, any row of Ω^- can be the eigenvector of $\mathbf{M}$ with eigenvalue -1 . Similarly, any row of Ω^+ can be the eigenvector of $\mathbf{M}$ with eigenvalue 1 .

Accordingly, there will be

1. any row of Ω^- is a linear combination of u_0 and u_1 .
2. any row of Ω^+ is a linear combination of u_2 and u_3 .

Consequently, $\text{rank}(\Omega^-) = \text{rank}(\Omega^+) = 2$; Ω^- and Ω^+ are not necessarily orthogonal matrices. Nonetheless, there is still $\Omega^- q = 0$, more specifically,

$$\Omega^- q = 0 \Leftrightarrow \begin{bmatrix} 0 & (a-b)^T \\ -(a-b) & [a+b]_{\times}^T \end{bmatrix} \begin{bmatrix} q_0 \\ q_1 \\ q_2 \\ q_3 \end{bmatrix} = 0 \quad (25)$$

$$\Leftrightarrow \begin{bmatrix} 0 & a_1-b_1 & a_2-b_2 & a_3-b_3 \\ -(a_1-b_1) & 0 & -(a_3+b_3) & a_2+b_2 \\ -(a_2-b_2) & a_3+b_3 & 0 & -(a_1+b_1) \\ -(a_3-b_3) & -(a_2+b_2) & a_1+b_1 & 0 \end{bmatrix} \begin{bmatrix} q_0 \\ q_1 \\ q_2 \\ q_3 \end{bmatrix} = 0 \quad (26)$$

5. Special Case II: Clifford Torus in $\mathbb{S}^3$

In the previous description, we provide a general derivation for the connection between rotation constraint, i.e., $b^T R a = 0$ and Clifford torus. Here we show an alternative way to obtain a Clifford torus from rotation constraint.

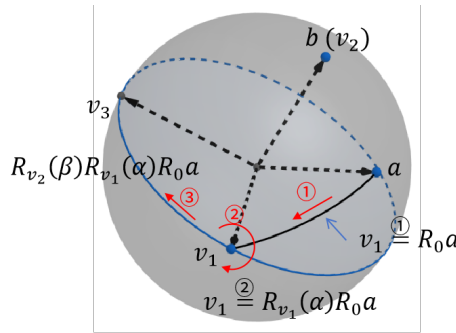


Figure 3. The geometric relationship of one rotation constraint $b^T R a = 0$ and its solution $R(\alpha, \beta) = R_{v_2}(\beta)R_{v_1}(\alpha)R_0$.

Given a rotation constraint $b^T R a = 0$, the possible rotation R can be (see Fig. 3)

$$R(\alpha, \beta) = R_{v_2}(\beta)R_{v_1}(\alpha)R_0, \quad \alpha, \beta \in [-\pi, \pi] \quad (27)$$

where $v_2 = b$, $v_1^T v_1 = 1$ and $v_1^T v_2 = 0$; R_0 is a rotation that rotates a to v_1 ; $R_{v_1}(\alpha) = \exp(\alpha[v_1]_{\times})$ and $R_{v_2}(\beta) = \exp(\beta[v_2]_{\times})$ are rotations around directions b_1 and b , respectively. The rotation motion can be as follows,

$$v_1 = R_0 a, \rightarrow v_1 = R_{v_1}(\alpha)R_0 a, \rightarrow b^T(R_{v_2}(\beta)R_{v_1}(\alpha)R_0 a) = 0 \quad (28)$$

To use quaternion to represent rotation [30]

$$\mathbf{R}_{v_2}(\beta) := \mathbf{q}_{v_2}(\beta) = \begin{bmatrix} \cos\left(\frac{\beta}{2}\right) & \mathbf{v}_2^T \sin\left(\frac{\beta}{2}\right) \end{bmatrix}^T \quad (29)$$

$$\mathbf{R}_{v_1}(\alpha) := \mathbf{q}_{v_1}(\alpha) = \begin{bmatrix} \cos\left(\frac{\alpha}{2}\right) & \mathbf{v}_1^T \sin\left(\frac{\alpha}{2}\right) \end{bmatrix}^T \quad (30)$$

$$\mathbf{R}_0 := \begin{bmatrix} q_0 & \mathbf{q}_{123}^T \end{bmatrix}^T \quad (31)$$

The formula for the multiplication of two quaternions is [30]

$$(r_1, \vec{v}_1)(r_2, \vec{v}_2) = (r_1 r_2 - \vec{v}_1 \cdot \vec{v}_2, r_1 \vec{v}_2 + r_2 \vec{v}_1 + \vec{v}_1 \times \vec{v}_2) \quad (32)$$

Define $\mathbf{v}_3 = \mathbf{v}_1 \times \mathbf{v}_2$, then

$$\begin{aligned} \mathbf{R}_{v_2}(\beta) \mathbf{R}_{v_1}(\alpha) &:= \\ &\begin{bmatrix} \cos\left(\frac{\beta}{2}\right) \cos\left(\frac{\alpha}{2}\right) & \mathbf{v}_2^T \sin\left(\frac{\beta}{2}\right) \cos\left(\frac{\alpha}{2}\right) + \mathbf{v}_1^T \sin\left(\frac{\alpha}{2}\right) \cos\left(\frac{\beta}{2}\right) - \mathbf{v}_3^T \sin\left(\frac{\beta}{2}\right) \sin\left(\frac{\alpha}{2}\right) \end{bmatrix}^T \end{aligned} \quad (33)$$

In addition, let $w_1 = \frac{1}{2}(\alpha - \beta)$ and $w_2 = \frac{1}{2}(\alpha + \beta)$,

$$\cos\left(\frac{\beta}{2}\right) \cos\left(\frac{\alpha}{2}\right) = \frac{1}{2} \left(\cos\left(\frac{\alpha - \beta}{2}\right) + \cos\left(\frac{\alpha + \beta}{2}\right) \right) = \frac{1}{2} (\cos(w_1) + \cos(w_2)) \quad (34)$$

$$\sin\left(\frac{\beta}{2}\right) \cos\left(\frac{\alpha}{2}\right) = \frac{1}{2} \left(\sin\left(\frac{\alpha + \beta}{2}\right) - \sin\left(\frac{\alpha - \beta}{2}\right) \right) = \frac{1}{2} (\sin(w_2) - \sin(w_1)) \quad (35)$$

$$\sin\left(\frac{\alpha}{2}\right) \cos\left(\frac{\beta}{2}\right) = \frac{1}{2} \left(\sin\left(\frac{\alpha + \beta}{2}\right) + \sin\left(\frac{\alpha - \beta}{2}\right) \right) = \frac{1}{2} (\sin(w_2) + \sin(w_1)) \quad (36)$$

$$\sin\left(\frac{\beta}{2}\right) \sin\left(\frac{\alpha}{2}\right) = \frac{1}{2} \left(\cos\left(\frac{\alpha - \beta}{2}\right) - \cos\left(\frac{\alpha + \beta}{2}\right) \right) = \frac{1}{2} (\cos(w_1) - \cos(w_2)) \quad (37)$$

Then,

$$\begin{aligned} \mathbf{R}_{v_2}(\beta) \mathbf{R}_{v_1}(\alpha) &:= \\ &\frac{1}{2} \begin{bmatrix} \cos(w_1) + \cos(w_2) & \sin(w_1)(\mathbf{v}_1^T - \mathbf{v}_2^T) - \cos(w_1)\mathbf{v}_3^T + \sin(w_2)(\mathbf{v}_1^T + \mathbf{v}_2^T) + \cos(w_2)\mathbf{v}_3^T \end{bmatrix}^T \end{aligned} \quad (38)$$

Therefore,

$$\mathbf{R}(\alpha, \beta) = \mathbf{R}_{v_2}(\beta) \mathbf{R}_{v_1}(\alpha) \mathbf{R}_0 \quad (39)$$

$$:= \frac{1}{2} \begin{bmatrix} k_0 & k_1 & k_2 & k_3 \end{bmatrix} \begin{bmatrix} \cos(w_1) \\ \sin(w_1) \\ \cos(w_2) \\ \sin(w_2) \end{bmatrix}, \quad w_1 = \frac{1}{2}(\alpha - \beta), \text{ and } w_2 = \frac{1}{2}(\alpha + \beta) \quad (40)$$

$$= \frac{1}{2} (k_0 \cos(w_1) + k_1 \sin(w_1) + k_2 \cos(w_2) + k_3 \sin(w_2)) \quad (41)$$

where

$$k_0 = \begin{bmatrix} q_0 + \mathbf{v}_3^T \mathbf{q}_{123} \\ \mathbf{q}_{123} - q_0 \mathbf{v}_3 - \mathbf{v}_3 \times \mathbf{q}_{123} \end{bmatrix} \quad k_1 = \begin{bmatrix} -(\mathbf{v}_1 - \mathbf{v}_2)^T \mathbf{q}_{123} \\ q_0(\mathbf{v}_1 - \mathbf{v}_2) + (\mathbf{v}_1 - \mathbf{v}_2) \times \mathbf{q}_{123} \end{bmatrix} \quad (42)$$

$$k_2 = \begin{bmatrix} q_0 - \mathbf{v}_3^T \mathbf{q}_{123} \\ \mathbf{q}_{123} + q_0 \mathbf{v}_3 + \mathbf{v}_3 \times \mathbf{q}_{123} \end{bmatrix} \quad k_3 = \begin{bmatrix} -(\mathbf{v}_1 + \mathbf{v}_2)^T \mathbf{q}_{123} \\ q_0(\mathbf{v}_1 + \mathbf{v}_2) + (\mathbf{v}_1 + \mathbf{v}_2) \times \mathbf{q}_{123} \end{bmatrix} \quad (43)$$

It is easy to verify

$$\begin{bmatrix} k_0 & k_1 & k_2 & k_3 \end{bmatrix}^T \begin{bmatrix} k_0 & k_1 & k_2 & k_3 \end{bmatrix} = \begin{bmatrix} 2 & 0 & 0 & 0 \\ 0 & 2 & 0 & 0 \\ 0 & 0 & 2 & 0 \\ 0 & 0 & 0 & 2 \end{bmatrix} \quad (44)$$

Geometrically, Eq. (41) describes a Clifford torus [51] in $\mathbb{S}^3$.

5.1. Intersections of Two Different Clifford Tori

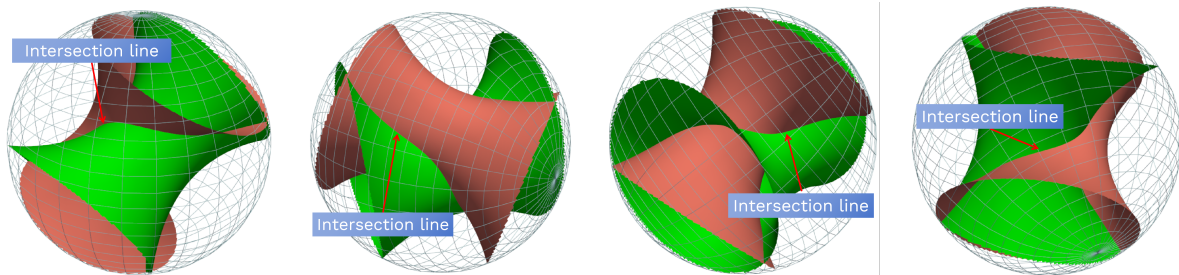


Figure 4. The illustration for intersection curves of two Clifford tori in $\mathbb{R}^3$ by stereographic projection.

Given two rotation constraints $\mathbf{b}_1^T \mathbf{R} \mathbf{a}_1 = 0$ and $\mathbf{b}_2^T \mathbf{R} \mathbf{a}_2 = 0$, the possible rotation can be considered in the intersections of two Clifford tori (see Fig. 4). Here, a solution to solve the intersection line is presented. Specifically,

$$\begin{cases} \mathbf{b}_1^T \mathbf{R} \mathbf{a}_1 = 0 \\ \mathbf{b}_2^T \mathbf{R} \mathbf{a}_2 = 0 \end{cases} \Rightarrow \begin{cases} \mathbf{R}(\alpha, \beta) = \mathbf{R}_{v_2}(\beta) \mathbf{R}_{v_1}(\alpha) \mathbf{R}_0 \\ \mathbf{b}_2^T \mathbf{R} \mathbf{a}_2 = 0 \end{cases} \Rightarrow \mathbf{b}_2^T \mathbf{R}_{v_2}(\beta) \mathbf{R}_{v_1}(\alpha) \mathbf{R}_0 \mathbf{a}_2 = 0 \quad (45)$$

where $\mathbf{v}_2 = \mathbf{b}_1$ and $\mathbf{v}_1 \in \mathbb{S}^2$ is an arbitrary vector and satisfies $\mathbf{v}_1^T \mathbf{v}_2 = 0$; $\mathbf{R}_0$ is a specific arbitrary rotation that can rotate $\mathbf{a}$ to $\mathbf{v}_1$; $\mathbf{R}_{v_1}(\alpha) = \exp(\alpha[\mathbf{v}_1]_{\times})$ and $\mathbf{R}_{v_2}(\beta) = \exp(\beta[\mathbf{v}_2]_{\times})$.

According to Rodrigues' rotation formula [3], we have

$$\mathbf{R}_{v_1}(\alpha) = \exp(\alpha[\mathbf{v}_1]_{\times}) = \mathbf{I} + \sin(\alpha)[\mathbf{v}_1]_{\times} + (1 - \cos(\alpha))[\mathbf{v}_1]_{\times}^2 \quad (46)$$

$$\mathbf{R}_{v_2}(\beta) = \exp(\beta[\mathbf{v}_2]_{\times}) = \mathbf{I} + \sin(\beta)[\mathbf{v}_2]_{\times} + (1 - \cos(\beta))[\mathbf{v}_2]_{\times}^2 \quad (47)$$

Therefore,

$$\mathbf{b}_2^T \left(\mathbf{I} + \sin(\beta)[\mathbf{v}_2]_{\times} + (1 - \cos(\beta))[\mathbf{v}_2]_{\times}^2 \right) \left(\mathbf{I} + \sin(\alpha)[\mathbf{v}_1]_{\times} + (1 - \cos(\alpha))[\mathbf{v}_1]_{\times}^2 \right) \mathbf{R}_0 \mathbf{a}_2 = 0 \quad (48)$$

Clearly, Eq.(48) shows a one-dimensional curve constructed by α and β (see Fig. 5).

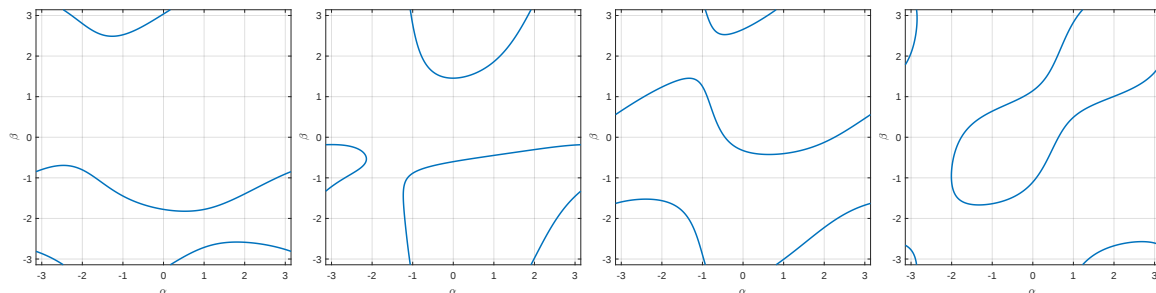


Figure 5. The illustration for intersection curves of two Clifford tori in α and β space.

Specifically, given a $\alpha \in [-\pi, \pi]$, β would be computed by Eq. (48). Intuitively, one should solve the following equation

$$\sin(\beta)A + \cos(\beta)B + C = 0 \quad (49)$$

where $A = \mathbf{b}_2^T [\mathbf{v}_2]_{\times} \mathbf{R}_{v_1}(\alpha) \mathbf{R}_0 \mathbf{a}_2$, $B = -\mathbf{b}_2^T [\mathbf{v}_2]_{\times}^2 \mathbf{R}_{v_1}(\alpha) \mathbf{R}_0 \mathbf{a}_2$ and $C = \mathbf{b}_2^T \mathbf{v}_2 \mathbf{v}_2^T \mathbf{R}_{v_1}(\alpha) \mathbf{R}_0 \mathbf{a}_2$. According to the tangent half-angle formula, when $\beta \neq \pm\pi$,

$$\frac{2 \tan\left(\frac{\beta}{2}\right)}{1 + \tan^2\left(\frac{\beta}{2}\right)} A + \frac{1 - \tan^2\left(\frac{\beta}{2}\right)}{1 + \tan^2\left(\frac{\beta}{2}\right)} B + C = 0$$

$$\xrightarrow{y \triangleq \tan\left(\frac{\beta}{2}\right)} 2Ay + (1 - y^2)B + (1 + y^2)C = 0 \Rightarrow (C - B)y^2 + 2Ay + B + C = 0 \quad (50)$$

Accordingly, β can be computed and rotation $\mathbf{R}(\alpha, \beta)$ can be computed as well. Formally,

$$f : \alpha \in [\pi, \pi] \xrightarrow{\beta \text{ by Eq.(48)}} \mathbf{R}(\alpha, \beta) \quad (51)$$

5.2. Intersections of Three Different Clifford Tori

Given three rotation constraints $\mathbf{b}_1^T \mathbf{R} \mathbf{a}_1 = 0$, $\mathbf{b}_2^T \mathbf{R} \mathbf{a}_2 = 0$ and $\mathbf{b}_3^T \mathbf{R} \mathbf{a}_3 = 0$, the possible rotation can be considered in the intersections of three Clifford tori. Note that it belongs to the famous E3Q3 problem [52], and there are many solutions have been proposed to solve it [53–57]. Here this paper provides a more straight linear way. Specifically,

$$\begin{cases} \mathbf{b}_1^T \mathbf{R} \mathbf{a}_1 = 0 \\ \mathbf{b}_2^T \mathbf{R} \mathbf{a}_2 = 0 \\ \mathbf{b}_3^T \mathbf{R} \mathbf{a}_3 = 0 \end{cases} \Rightarrow \begin{cases} \mathbf{R}(\alpha, \beta) = \mathbf{R}_{v_2}(\beta) \mathbf{R}_{v_1}(\alpha) \mathbf{R}_0 \\ \mathbf{b}_2^T \mathbf{R} \mathbf{a}_2 = 0 \\ \mathbf{b}_3^T \mathbf{R} \mathbf{a}_3 = 0 \end{cases} \Rightarrow \begin{cases} \mathbf{b}_2^T \mathbf{R}_{v_2}(\beta) \mathbf{R}_{v_1}(\alpha) \mathbf{R}_0 \mathbf{a}_2 = 0 \\ \mathbf{b}_3^T \mathbf{R}_{v_2}(\beta) \mathbf{R}_{v_1}(\alpha) \mathbf{R}_0 \mathbf{a}_3 = 0 \end{cases} \quad (52)$$

Therefore,

$$\mathbf{b}_2^T \left(\mathbf{I} + \sin(\beta) [\mathbf{v}_2]_{\times} + (1 - \cos(\beta)) [\mathbf{v}_2]_{\times}^2 \right) \left(\mathbf{I} + \sin(\alpha) [\mathbf{v}_1]_{\times} + (1 - \cos(\alpha)) [\mathbf{v}_1]_{\times}^2 \right) \mathbf{R}_0 \mathbf{a}_2 = 0 \quad (53)$$

$$\mathbf{b}_3^T \left(\mathbf{I} + \sin(\beta) [\mathbf{v}_2]_{\times} + (1 - \cos(\beta)) [\mathbf{v}_2]_{\times}^2 \right) \left(\mathbf{I} + \sin(\alpha) [\mathbf{v}_1]_{\times} + (1 - \cos(\alpha)) [\mathbf{v}_1]_{\times}^2 \right) \mathbf{R}_0 \mathbf{a}_3 = 0 \quad (54)$$

There are two unknown variables α and β for two constraints, and the intersections can be computed. According to tangent half-angle formula (a.k.a. Weierstrass substitution formulas),

$$\sin(\alpha) = \frac{2 \tan\left(\frac{\alpha}{2}\right)}{1 + \tan^2\left(\frac{\alpha}{2}\right)}, \quad \cos(\alpha) = \frac{1 - \tan^2\left(\frac{\alpha}{2}\right)}{1 + \tan^2\left(\frac{\alpha}{2}\right)}, \quad (55)$$

$$\sin(\beta) = \frac{2 \tan\left(\frac{\beta}{2}\right)}{1 + \tan^2\left(\frac{\beta}{2}\right)}, \quad \cos(\beta) = \frac{1 - \tan^2\left(\frac{\beta}{2}\right)}{1 + \tan^2\left(\frac{\beta}{2}\right)} \quad (56)$$

Let $x = \tan\left(\frac{\alpha}{2}\right)$, and $y = \tan\left(\frac{\beta}{2}\right)$, and when $\alpha \neq \pm\pi$, $\beta \neq \pm\pi$,

$$\mathbf{b}_2^T \left((1 + y^2) \mathbf{I} + 2y [\mathbf{v}_2]_{\times} + 2y^2 [\mathbf{v}_2]_{\times}^2 \right) \left((1 + x^2) \mathbf{I} + 2x [\mathbf{v}_1]_{\times} + 2x^2 [\mathbf{v}_1]_{\times}^2 \right) \mathbf{R}_0 \mathbf{a}_2 = 0 \quad (57)$$

$$\mathbf{b}_3^T \left((1 + y^2) \mathbf{I} + 2y [\mathbf{v}_2]_{\times} + 2y^2 [\mathbf{v}_2]_{\times}^2 \right) \left((1 + x^2) \mathbf{I} + 2x [\mathbf{v}_1]_{\times} + 2x^2 [\mathbf{v}_1]_{\times}^2 \right) \mathbf{R}_0 \mathbf{a}_3 = 0 \quad (58)$$

The system of equations can be reformulated as

$$\mathcal{F}(x, y) = \begin{bmatrix} y^2 & y & 1 \end{bmatrix} \mathbf{A} \begin{bmatrix} x^2 \\ x \\ 1 \end{bmatrix} = 0, \quad \mathcal{G}(x, y) = \begin{bmatrix} y^2 & y & 1 \end{bmatrix} \mathbf{B} \begin{bmatrix} x^2 \\ x \\ 1 \end{bmatrix} = 0 \quad (59)$$

where $\mathbf{A}$ and $\mathbf{B} \in \mathbb{R}^{3 \times 3}$, Specifically,

$$\mathbf{A} = \begin{bmatrix} A_{1,1} & A_{1,2} & A_{1,3} \\ A_{2,1} & A_{2,2} & A_{2,3} \\ A_{3,1} & A_{3,2} & A_{3,3} \end{bmatrix} = \mathbf{b}_2^T * \begin{bmatrix} (\mathbf{I} + 2[\mathbf{v}_2]_{\times}^2)(\mathbf{I} + 2[\mathbf{v}_1]_{\times}^2) & (\mathbf{I} + 2[\mathbf{v}_2]_{\times}^2)(2[\mathbf{v}_1]_{\times}) & \mathbf{I} + 2[\mathbf{v}_2]_{\times}^2 \\ (2[\mathbf{v}_2]_{\times})(\mathbf{I} + 2[\mathbf{v}_1]_{\times}^2) & (2[\mathbf{v}_2]_{\times})(2[\mathbf{v}_1]_{\times}) & 2[\mathbf{v}_2]_{\times} \\ \mathbf{I} + 2[\mathbf{v}_1]_{\times}^2 & 2[\mathbf{v}_1]_{\times} & \mathbf{I} \end{bmatrix} * \mathbf{R}_0 \mathbf{a}_2 \quad (60)$$

$$\mathbf{B} = \begin{bmatrix} B_{1,1} & B_{1,2} & B_{1,3} \\ B_{2,1} & B_{2,2} & B_{2,3} \\ B_{3,1} & B_{3,2} & B_{3,3} \end{bmatrix} = \mathbf{b}_3^T * \begin{bmatrix} (\mathbf{I} + 2[\mathbf{v}_2]_{\times}^2)(\mathbf{I} + 2[\mathbf{v}_1]_{\times}^2) & (\mathbf{I} + 2[\mathbf{v}_2]_{\times}^2)(2[\mathbf{v}_1]_{\times}) & \mathbf{I} + 2[\mathbf{v}_2]_{\times}^2 \\ (2[\mathbf{v}_2]_{\times})(\mathbf{I} + 2[\mathbf{v}_1]_{\times}^2) & (2[\mathbf{v}_2]_{\times})(2[\mathbf{v}_1]_{\times}) & 2[\mathbf{v}_2]_{\times} \\ \mathbf{I} + 2[\mathbf{v}_1]_{\times}^2 & 2[\mathbf{v}_1]_{\times} & \mathbf{I} \end{bmatrix} * \mathbf{R}_0 \mathbf{a}_3 \quad (61)$$

and $*$ is to calculate the matrix item by item. Geometrically, if only considering real solutions, it describes the intersections of two curves computed by observations in $z = 0$ plane (see Fig.6).

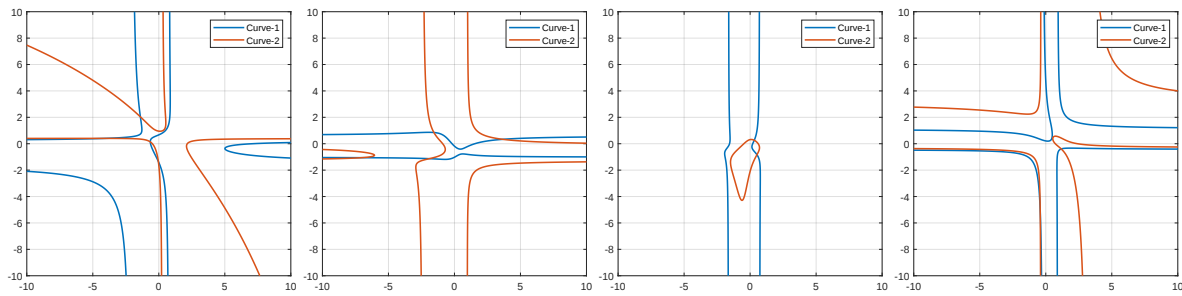


Figure 6. The intersection of two curves that are constructed by rotation constraints in the 2D plane. Specifically, we can consider $z = \mathcal{F}(x, y)$ and $z = \mathcal{G}(x, y)$ as two surfaces in $\mathbb{R}^3$. Let $z = 0$, it means $z = 0$ plane and it cuts the surfaces $\mathcal{F}(x, y)$ and $\mathcal{G}(x, y)$. Consequently, there are two intersection lines, which can be drawn in $z = 0$ plane.

Algebraically, it is a polynomial system containing two equations in two variables, which can be generally solved by resultant-based method [58,59]. We denote

$$\begin{bmatrix} y^2 & y & 1 \end{bmatrix} \mathbf{A} \begin{bmatrix} x^2 \\ x \\ 1 \end{bmatrix} = \mathcal{A}(y) \begin{bmatrix} x^2 \\ x \\ 1 \end{bmatrix} = 0, \quad \begin{bmatrix} y^2 & y & 1 \end{bmatrix} \mathbf{B} \begin{bmatrix} x^2 \\ x \\ 1 \end{bmatrix} = \mathcal{B}(y) \begin{bmatrix} x^2 \\ x \\ 1 \end{bmatrix} = 0 \quad (62)$$

Both sides of the equations are multiplied by x ,

$$\mathcal{A}(y) \begin{bmatrix} x^3 \\ x^2 \\ x \end{bmatrix} = 0, \quad \mathcal{B}(y) \begin{bmatrix} x^3 \\ x^2 \\ x \end{bmatrix} = 0 \quad (63)$$

By stacking equations,

$$\begin{bmatrix} 0 & \mathcal{A}(y) \\ 0 & \mathcal{B}(y) \\ \mathcal{A}(y) & 0 \\ \mathcal{B}(y) & 0 \end{bmatrix} \begin{bmatrix} x^3 \\ x^2 \\ x \\ 1 \end{bmatrix} = \mathcal{H}(y) \begin{bmatrix} x^3 \\ x^2 \\ x \\ 1 \end{bmatrix} = 0 \quad (64)$$

To solve y , it is to solve $|\mathcal{H}(y)| = 0$. This determinant is also known as a hidden-variable resultant in computer vision filed [60,61], and it is an univariate polynomial with degree 8 of the hidden variable y .

$$|\mathcal{H}(y)| = a_8y^8 + a_7y^7 + a_6y^6 + a_5y^5 + a_4y^4 + a_3y^3 + a_2y^2 + a_1y + a_0 = 0 \quad (65)$$

In fact, the problem is not very complex, one can directly calculate the explicit expression of each term for the resultant-based method using symbolic math. Practically, one can obtain explicit coefficients $\{a_i\}_{i=0}^8$ by the function `resultant` in Matlab¹ or the function `sympy.polys.polytools.resultant` in SymPy², and this paper will not list all details to avoid long-and-tedious expressions for readability. Nonetheless, the solution of y can be directly computed by solving roots of a univariate polynomial [62, 63]. After solving y , it is easy to calculate x .

5.2.1. Linear Solution to Solve Intersections of Three Different Clifford Tori

Alternatively, Eq. (64) can be rewritten as a quadratic eigenvalue problem, i.e., $(\lambda^2\mathbf{H}_2 + \lambda\mathbf{H}_1 + \mathbf{H}_0)v = 0$ [60,61,64].

$$\mathcal{H}(y) \begin{bmatrix} x^3 \\ x^2 \\ x \\ 1 \end{bmatrix} = 0 \Rightarrow (y^2\mathbf{H}_2 + y\mathbf{H}_1 + \mathbf{H}_0) \begin{bmatrix} x^3 \\ x^2 \\ x \\ 1 \end{bmatrix} = 0 \quad (66)$$

where $\mathbf{H}_0, \mathbf{H}_1$ and $\mathbf{H}_2 \in \mathbb{R}^{4 \times 4}$. Specifically,

$$\mathbf{H}_2 = \begin{bmatrix} 0 & A_{1,1} & A_{1,2} & A_{1,3} \\ 0 & B_{1,1} & B_{1,2} & B_{1,3} \\ A_{1,1} & A_{1,2} & A_{1,3} & 0 \\ B_{1,1} & B_{1,2} & B_{1,3} & 0 \end{bmatrix} \quad \mathbf{H}_1 = \begin{bmatrix} 0 & A_{2,1} & A_{2,2} & A_{2,3} \\ 0 & B_{2,1} & B_{2,2} & B_{2,3} \\ A_{2,1} & A_{2,2} & A_{2,3} & 0 \\ B_{2,1} & B_{2,2} & B_{2,3} & 0 \end{bmatrix}$$

$$\mathbf{H}_0 = \begin{bmatrix} 0 & A_{3,1} & A_{3,2} & A_{3,3} \\ 0 & B_{3,1} & B_{3,2} & B_{3,3} \\ A_{3,1} & A_{3,2} & A_{3,3} & 0 \\ B_{3,1} & B_{3,2} & B_{3,3} & 0 \end{bmatrix}$$

Notably, there is no x and y in $\mathbf{H}_2, \mathbf{H}_1$ and $\mathbf{H}_0$. Accordingly, it is easy to solve the system of polynomial equations by linear way [64].

Specifically, the to-be-solved eigenvector of this problem is $v = [x^3 \ x^2 \ x \ 1]^T$ and corresponding eigenvalue is $\lambda = y$. Furthermore, the quadratic eigenvalue problem can be transformed into a generalized eigenvalue problem, i.e., $\mathbf{A}x = \lambda\mathbf{B}x$ [65], which has been extensively studied [61,65–67]. Specifically,

$$(\lambda^2\mathbf{H}_2 + \lambda\mathbf{H}_1 + \mathbf{H}_0)v = 0 \Rightarrow \begin{bmatrix} \mathbf{0} & \mathbf{I} \\ -\mathbf{H}_0 & -\mathbf{H}_1 \end{bmatrix} \begin{bmatrix} v \\ \lambda v \end{bmatrix} = \lambda \begin{bmatrix} \mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{H}_2 \end{bmatrix} \begin{bmatrix} v \\ \lambda v \end{bmatrix} \quad (67)$$

Therefore, there will be 8 eigenvalues of this linear problem. The easiest way to solve this problem is using the Matlab function built-in `polyeig`³. SciPy also provides a function `scipy.linalg.eig`⁴ to solve generalized eigenvalue problems.

Conflicts of Interest: The author declares no conflicts of interest.

References

1. Waldron, K.J.; Schmiedeler, J. Kinematics. *Springer handbook of robotics* **2016**, pp. 11–36.

¹ <https://www.mathworks.com/help/symbolic/sym.resultant.html>

² <https://www.sympy.org/en/index.html>

³ <https://www.mathworks.com/help/matlab/ref/polyeig.html>

⁴ <https://docs.scipy.org/doc/scipy/reference/generated/scipy.linalg.eigvals.html#scipy.linalg.eigvals>

2. Ivanov, A. Theoretical matrix study of rigid body general motion. *Greener Journal of Physics and Natural Sciences* **2017**, 3, 009–020.
3. Hartley, R.; Zisserman, A. *Multiple view geometry in computer vision*; Cambridge university press, 2003.
4. Yehia, H.M. Rigid Body Dynamics. *Advances in Mechanics and Mathematics. Birkhäuser Cham* **2022**.
5. Teisseyre, R. Tutorial on new developments in the physics of rotational motions. *Bulletin of the Seismological Society of America* **2009**, 99, 1028–1039.
6. Kisantal, M.; Sharma, S.; Park, T.H.; Izzo, D.; Märten, M.; D'Amico, S. Satellite pose estimation challenge: Dataset, competition design, and results. *IEEE Transactions on Aerospace and Electronic Systems* **2020**, 56, 4083–4098.
7. Gul, F.; Rahiman, W.; Nazli Alhady, S.S. A comprehensive study for robot navigation techniques. *Cogent Engineering* **2019**, 6, 1632046.
8. Yefymenko, N.; Kudermetov, R. Quaternion models of a rigid body rotation motion and their application for spacecraft attitude control. *Acta Astronautica* **2022**, 194, 76–82.
9. Rizzi, G.; Ruggiero, M.L. *Relativity in rotating frames: relativistic physics in rotating reference frames*; Springer, 2004.
10. Bustos, A.P.; Chin, T.J. Guaranteed Outlier Removal for Point Cloud Registration with Correspondences. *IEEE transactions on pattern analysis and machine intelligence* **2018**, 40, 2868–2882.
11. Eriksson, A.; Olsson, C.; Kahl, F.; Chin, T.J. Rotation Averaging with the Chordal Distance: Global Minimizers and Strong Duality. *IEEE Transactions on Pattern Analysis & Machine Intelligence* **2021**, 43, 256–268.
12. Crassidis, J.L.; Markley, F.L.; Cheng, Y. Survey of nonlinear attitude estimation methods. *Journal of guidance, control, and dynamics* **2007**, 30, 12–28.
13. Carlone, L.; et al. Estimation Contracts for Outlier-Robust Geometric Perception. *Foundations and Trends® in Robotics* **2023**, 11, 90–224.
14. Mathavaraj, S.; Butcher, E.A. SE (3)-constrained extended Kalman filtering for rigid body pose estimation. *IEEE Transactions on Aerospace and Electronic Systems* **2021**, 58, 2482–2492.
15. Welch, G.F. Kalman filter. *Computer Vision: A Reference Guide* **2020**, pp. 1–3.
16. Trawny, N.; Roumeliotis, S.I. Indirect Kalman filter for 3D attitude estimation. *University of Minnesota, Dept. of Comp. Sci. & Eng., Tech. Rep* **2005**, 2, 2005.
17. Chin, T.J.; Suter, D. *The maximum consensus problem: recent algorithmic advances*; Springer Nature, 2022.
18. Etingof, P. Lie groups and Lie algebras. *arXiv preprint arXiv:2201.09397* **2022**.
19. Dehmamy, N.; Walters, R.; Liu, Y.; Wang, D.; Yu, R. Automatic symmetry discovery with lie algebra convolutional network. *Advances in Neural Information Processing Systems* **2021**, 34, 2503–2515.
20. Gill, P.E.; Murray, W.; Wright, M.H. *Numerical linear algebra and optimization*; SIAM, 2021.
21. Farin, G.; Hansford, D. *Practical linear algebra: a geometry toolbox*; Chapman and Hall/CRC, 2021.
22. Horn, R.A.; Johnson, C.R. *Matrix analysis*; Cambridge university press, 2012.
23. Taubin, G. 3D Rotations. *IEEE Computer Graphics and Applications* **2011**, 31, 84–89. <https://doi.org/10.1109/MCG.2011.92>.
24. Ball, R.S. The theory of screws: A study in the dynamics of a rigid body. *Mathematische Annalen* **1876**, 9, 541–553.
25. Palais, B.; Palais, R. Euler's fixed point theorem: The axis of a rotation. *Journal of fixed point theory and applications* **2007**, 2, 215–220.
26. Kumar, V. The Theorems of Euler and Chasles. *University of Pennsylvania School of Engineering and Applied Science, I-Net, USA* **2000**.
27. Joseph, T. An alternative proof of Euler's rotation theorem. *The Mathematical Intelligencer* **2020**, 42, 44–49.
28. Fraiture, L. A history of the description of the three-dimensional finite rotation. *The Journal of the Astronautical Sciences* **2009**, 57, 207–232.
29. Hartley, R.I.; Kahl, F. Global optimization through rotation space search. *International Journal of Computer Vision* **2009**, 82, 64–79.
30. Jia, Y.B. Quaternions and rotations. *Com S* **2008**, 477, 15.
31. Spring, K.W. Euler parameters and the use of quaternion algebra in the manipulation of finite rotations: a review. *Mechanism and machine theory* **1986**, 21, 365–373.
32. Bauchau, O.; Bauchau, O. Parameterization of rotation. *Flexible Multibody Dynamics* **2011**, pp. 511–542.
33. Hartley, R.; Trunpf, J.; Dai, Y.; Li, H. Rotation averaging. *International journal of computer vision* **2013**, 103, 267–305.
34. Kim, S.; Kim, M. Rotation representations and their conversions. *IEEE Access* **2023**, 11, 6682–6699.

35. Shuster, M.D. A survey of attitude representation. *Journal of The Astronautical Sciences* **1993**, *41*, 439–517.
36. Liu, Y. Globally Optimal Solutions for Unit-Norm Constrained Computer Vision Problems. PhD thesis, Technische Universität München, 2022.
37. Liu, Y.; Dong, Y.; Song, Z.; Wang, M. 2D-3D Point Set Registration Based on Global Rotation Search. *IEEE Transactions on Image Processing* **2019**, *28*, 2599–2613.
38. Yang, J.; Li, H.; Campbell, D.; Jia, Y. Go-ICP: A globally optimal solution to 3D ICP point-set registration. *IEEE transactions on pattern analysis and machine intelligence* **2015**, *38*, 2241–2254.
39. Altmann, S.L. *Rotations, quaternions, and double groups*; Courier Corporation, 2005.
40. HORN, B.P. Closed-form solution of absolute orientation using unit quaternions. *Journal of the Optical Society of America. A, Optics and image science* **1987**, *4*, 629–642.
41. Peng, L.; Tsakiris, M.C.; Vidal, R. Arcs: Accurate rotation and correspondence search. In Proceedings of the Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2022, pp. 11153–11163.
42. McClintock, P. Relating plane transformations with stereographic projection. PhD thesis, 2022.
43. Wilkins, D.R. Möbius transformations and stereographic projection, 2017.
44. Gluck, H.; Warner, F.W. Great circle fibrations of the three-sphere. *Duke Mathematical Journal* **1983**, *50*, 107 – 132. <https://doi.org/10.1215/S0012-7094-83-05003-2>.
45. Bayro-Corrochano, E. A survey on quaternion algebra and geometric algebra applications in engineering and computer science 1995–2020. *IEEE Access* **2021**, *9*, 104326–104355.
46. Nastro, V.; Tancredi, U. Great circle navigation with vectorial methods. *The Journal of navigation* **2010**, *63*, 557–563.
47. Cheng, Q.M.; Ishikawa, S. A characterization of the Clifford torus. *Proceedings of the American Mathematical Society* **1999**, *127*, 819–828.
48. Yu, T.; Chen, J. Uniqueness of Clifford torus with prescribed isoperimetric ratio. *Proceedings of the American Mathematical Society* **2022**, *150*, 1749–1765.
49. Srivatsan, R.A.; Rosen, G.T.; Mohamed, D.F.N.; Choset, H. Estimating SE (3) elements using a dual quaternion based linear Kalman filter. In Proceedings of the Robotics: Science and systems, 2016.
50. Arun Srivatsan, R.; Zevallos, N.; Vagdargi, P.; Choset, H. Registration with a small number of sparse measurements. *The International Journal of Robotics Research* **2019**, *38*, 1403–1419.
51. Andersson, O.; Bengtsson, I. Clifford tori and unbiased vectors. *Reports on mathematical physics* **2017**, *79*, 33–51.
52. Kukulova, Z.; Heller, J.; Fitzgibbon, A. Efficient intersection of three quadrics and applications in computer vision. In Proceedings of the Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 1799–1808.
53. Xu, C.; Zhang, L.; Cheng, L.; Koch, R. Pose estimation from line correspondences: A complete analysis and a series of solutions. *IEEE transactions on pattern analysis and machine intelligence* **2016**, *39*, 1209–1222.
54. Yu, Q.; Xu, G.; Cheng, Y. An efficient and globally optimal method for camera pose estimation using line features. *Machine Vision and Applications* **2020**, *31*, 1–11.
55. Wang, P.; Xu, G.; Cheng, Y.; Yu, Q. Camera pose estimation from lines: a fast, robust and general method. *Machine Vision and Applications* **2019**, *30*, 603–614.
56. Chan, K. A simple mathematical approach for determining intersection of quadratic surfaces. In *Multiscale optimization methods and applications*; Springer, 2006; pp. 271–298.
57. Sarraga, R.F. Algebraic methods for intersections of quadric surfaces in GMSOLID. *Computer Vision, Graphics, and Image Processing* **1983**, *22*, 222–238.
58. Sturmfels, B. Introduction to resultants. In Proceedings of the Proceedings of Symposia in Applied Mathematics. American Mathematical Society, 1998, Vol. 53, pp. 25–40.
59. Woody, H. Polynomial resultants. *GNU operating system* **2016**.
60. Hartley, R.; Li, H. An efficient hidden variable approach to minimal-case camera motion estimation. *IEEE transactions on pattern analysis and machine intelligence* **2012**, *34*, 2303–2314.
61. Kukulova, Z.; Bujnak, M.; Pajdla, T. Polynomial eigenvalue solutions to minimal problems in computer vision. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **2011**, *34*, 1381–1393.
62. Pan, V.Y. Good News for Polynomial Root-finding. *arXiv preprint arXiv:1805.12042* **2018**.
63. Kalantari, B. *Polynomial root-finding and polynomiography*; World Scientific, 2008.
64. Tisseur, F.; Meerbergen, K. The quadratic eigenvalue problem. *SIAM review* **2001**, *43*, 235–286.

65. Ghojogh, B.; Karray, F.; Crowley, M. Eigenvalue and Generalized Eigenvalue Problems: Tutorial, 2023, [[arXiv:stat.ML/1903.11240](https://arxiv.org/abs/1903.11240)].
66. Williams, M.P. Solving polynomial equations using linear algebra. *Johns Hopkins APL Technical Digest* **2010**, *28*, 354–363.
67. Lancaster, P.; Zaballa, I. Diagonalizable quadratic eigenvalue problems. *Mechanical Systems and Signal Processing* **2009**, *23*, 1134–1144.

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.