Orientation, Rotation, Velocity, and Acceleration and the SRM

Version 1.01

13 June 2007

Paul Berner, PhD

Table of contents

1 Overview

Direction, orientation, and rotation are defined for 3D Euclidean space. Various representations of a rotation and the conversion between representations are presented. The SRM (ISO/IEC 18026:2006(E)) definition of direction in either a linear or curvilinear spatial reference frame is presented, and the conversion of a direction representation from one spatial reference frame to another spatial reference frame is discussed.

Many concepts discussed here have been in wide use from the time of Euler's work on the subject. As a result, there are many similar but different treatments in the literature. In particular, there are similar terms with different meanings and, in some cases, the differences are subtle. There are also many differences in notation conventions. For this reason an attempt has been made to provide self contained derivations (assuming the prerequisites) of most of the formulations presented here. By following the derivations there should be no mistake as to the intended meanings of the results. To improve the flow of the text, parts of lengthier derivations have been relegated to appendices.

1.1 Prerequisites

This document assumes that reader is familiar with linear algebra, calculus and elementary physics. The prerequisites include:

- Vector spaces,
    - vector dot and cross products
- Matrix algebra,
- Calculus
- Physics (Rigid body kinematics)

1.2 Notation

The coordinate representation of a three dimensional (3D) vector $\boldsymbol{u}$ with respect to basis is a column vector $\boldsymbol{u} = \begin{pmatrix} u_1 \\ u_2 \\ u_3 \end{pmatrix}$. To compactly denote a coordinate in a line of text, the transpose is used $\boldsymbol{u} = (u_1, u_2, u_3)^{\mathsf{T}}$.

Inner product or *dot product* or *scalar product* of 3D vectors $\boldsymbol{u} = (u_1, u_2, u_3)^{\mathsf{T}}$ and $\boldsymbol{v} = (v_1, v_2, v_3)^{\mathsf{T}}$ is defined as:

$$\boldsymbol{u} \bullet \boldsymbol{v} = \boldsymbol{u}^{\mathsf{T}} \boldsymbol{v} = u_1 v_1 + u_2 v_2 + u_3 v_3 \tag{0.1}$$

The *outer product* of 3D vectors $\boldsymbol{u}$ and $\boldsymbol{v}$ is defined as:

$$\boldsymbol{u} \otimes \boldsymbol{v} = \boldsymbol{u}\boldsymbol{v}^{\mathsf{T}} = \begin{pmatrix} u_1v_1 & u_1v_2 & u_1v_3 \\ u_2v_1 & u_2v_2 & u_2v_3 \\ u_3v_1 & u_3v_2 & u_3v_3 \end{pmatrix} \tag{0.2}$$

Note that:

$$\text{Trace}\left(\boldsymbol{u} \otimes \boldsymbol{v}\right) = u_1v_1 + u_2v_2 + u_3v_3 = \boldsymbol{u} \bullet \boldsymbol{v} \tag{0.3}$$

The *vector product* or *cross product* of 3D vectors $\boldsymbol{u}$ and $\boldsymbol{v}$ is defined as:

$$\boldsymbol{u} \times \boldsymbol{v} = \left(u_2v_3 - u_3v_2,\, u_3v_1 - u_1v_3,\, u_1v_2 - u_2v_1\right)^{\mathsf{T}} = \boldsymbol{S_u}\boldsymbol{v} \tag{0.4}$$

where:

$$\boldsymbol{S_u} = \begin{pmatrix} 0 & -u_3 & +u_2 \\ +u_3 & 0 & -u_1 \\ -u_2 & +u_1 & 0 \end{pmatrix}$$

See Appendix A for some cross product properties.

The *norm* or *length of a vector* $\boldsymbol{u}$ is defined as:

$$\|\boldsymbol{u}\| = \sqrt{\boldsymbol{u} \bullet \boldsymbol{u}} \tag{0.5}$$

The 3D *identity matrix* is denoted as:

$$\boldsymbol{I}_{3\times 3} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \tag{0.6}$$

The 3D *zero vector* is noted as:

$$\mathbf{0} = (0,0,0)^{\mathsf{T}} \tag{0.7}$$

The two argument form of inverse tangent, $\text{arctan}2\left(y,x\right)$, returns a value adjusted by the quadrant of the point $\left(x,y\right)$. Given real numbers x and y,

$\arctan 2(y,x)=\theta$

where: θ is the unique value satisying $-\pi<\theta\le\pi$, and

if $r=0$,

$\theta=0$, else

if $r>0$,

$x=r\cos\theta$ and $y=r\sin\theta$.

where:

$r=\sqrt{x^2+y^2}$.

2 Vectors, directions, axes and their uses

2.1 Vector space directions

A direction in a Euclidean vector space may be represented as a *unit vector*. That is, a vector $\boldsymbol{n}$ of length 1. Any non-zero $\boldsymbol{u}$ vector may be *normalized* to a unit vector $\boldsymbol{n}$ by dividing by the norm of the vector:

$$\boldsymbol{n}=\frac{1}{\|\boldsymbol{u}\|}\boldsymbol{u},\quad \text{if } \boldsymbol{u}\ne(0,0,0)^{\mathrm{T}}.$$

Any positive multiple of a unit vector points in the same direction. By requiring unit vectors, each direction has a unique vector representation.

Directions have many application specific uses. For example, a velocity is a direction multiplied by a speed. Force and momentum acting on the center of mass of a body may be similarly represented.

A direction can be used to specify the axis of a rotating body. The axis of a rotating body lies on a line. By specifying the line as a direction, the right hand rule can be used to unambiguously identify which of the two axial rotational directions is acting on the body. Torque and angular momentum acting on a body may be similarly represented.

2.2 Vector directions in the SRM

In the Spatial Reference Model (SRM), the underlying vector space that is associated with a 3D Spatial Reference Frame (SRF) is determined by the Object Reference Model (ORM) of the SRF. For example, the 3D vector space of any 3D SRF based on ORM WGS84 corresponds to the geocentric SRF for ORM WGS84. This associated 3D Euclidean space is called the *object-space* of the ORM.

An SRF associates unique coordinates in a domain of coordinate-space to corresponding points in object-space. In the special case of a geocentric SRF, the object-space and coordinate-space are indistinguishable. In general, an SRF is either linear or curvilinear. In the linear cases, the vector-space structure of coordinate-space carries over to object-space. In particular, lines through points in a given direction $\boldsymbol{n}$ are all parallel in both coordinate- and object-space. This shows that a direction is translation invariant in a linear SRF. A linear SRF will not preserve angular relationships between directions unless the associated CS is also orthonormal. In the orthonormal case, angles and distances are preserved.

In general, there is no vector-space structure in the coordinate space of a curvilinear SRF. The coordinate-space of an augmented map projection SRF (a map projection augmented with ellipsoidal height as a third dimension) appears to inherit the vector-space structure of $\mathbf{R}^3$, however, the vector properties of (easting, northing, height) coordinates do not carry over to object-space. This is illustrated in part by the "up pointing" vector $\boldsymbol{n}$ = (0, 0, 1) that points in different spatial directions (in object-space) depending on the map coordinate location from which $\boldsymbol{n}$ is viewed.

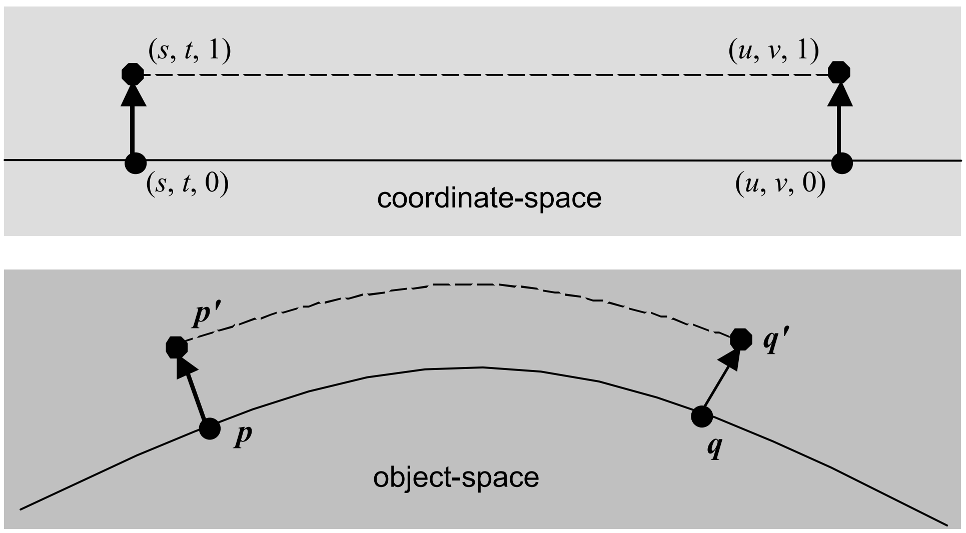

Figure 1 – Directions in an augmented map projection SRF

In Figure 1, distinct position points $\boldsymbol{p}$ and $\boldsymbol{q}$ on the ellipsoid surface are projected to augmented map coordinates $(s, t, 0)$ and $(u, v, 0)$. Starting at these map coordinates, the coordinates one unit away in direction $\boldsymbol{n}$ are $(s, t, 1)$ and $(u, v, 1)$ respectively. In an augmented map projection, these coordinates correspond to the position-space points $\boldsymbol{p}'$ and $\boldsymbol{q}'$. The direction from $\boldsymbol{p}$ to $\boldsymbol{p}'$ is not the same as the direction from $\boldsymbol{q}$ to $\boldsymbol{q}'$. This shows that the "up direction" is relative to an observation or reference point.

For each reference point, the SRM defines a uniform method for associating a unique orthonormal linear SRF to each reference point coordinate. This associated linear SRF will be used to specify a direction as "seen" from the reference point. This SRF is called the *local tangent frame* at the reference point. This SRF is defined by as having its origin at the reference point and axis directions given by the normalized tangent vectors to the coordinate curves passing through the reference point as illustrated in Figure 2. All curvilinear SRFs in the SRM are orthogonal so that the local tangent frame will be an orthonormal linear SRF.

Continuing the augmented map projection example, Figure 3 shows the local tangent frames axes (x and z axes) at points $\boldsymbol{p}$ and $\boldsymbol{q}$. The local "up" directions may be specified in either local tangent frame. Since directions are translation invariant in linear SRFs, we may conceptually translate the two local tangent frames to a common origin as in Figure 4.

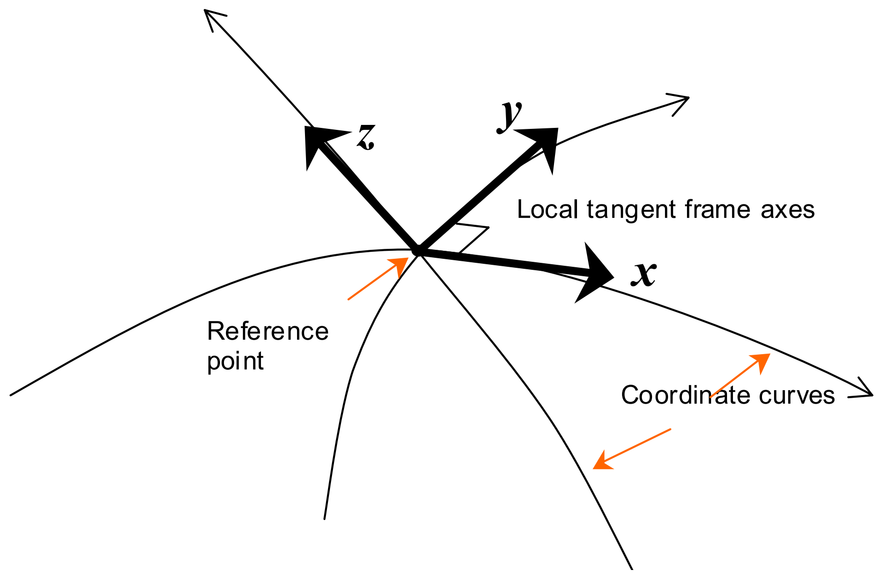

Figure 2 – Local tangent frame axes

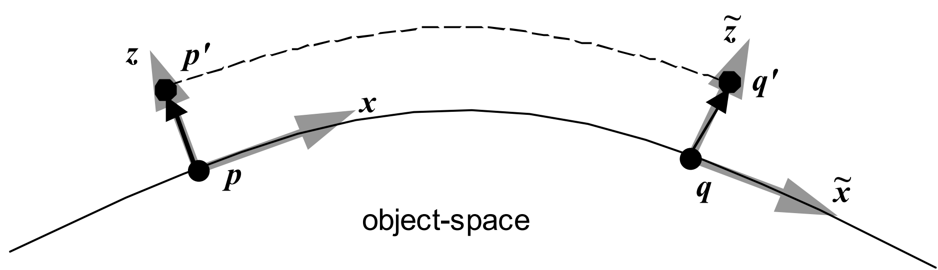

Figure 3 – Local tangent frame axes at p and q

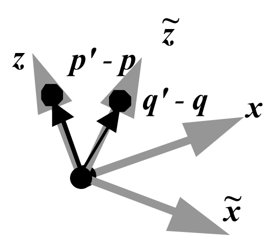

Figure 4 – Direction vectors two local tangent frames

In the SRM, a Direction data type consists of a reference point coordinate in a given SRF and a 3-tuple unit vector in the local tangent frame at the reference point. Since there is neither an intrinsic SRF nor an intrinsic reference point to specify a general direction in object-space, it is necessary to be able to inter-convert between SRF and reference point specifications of a given direction. The SRM approach of associating reference points and local tangent frames reduces the general problem to the problem of inter-converting between two orthonormal linear spaces. This problem generalizes to the problem of inter-converting any vector quantity between a pair of linear spaces. The treatment of this problem begins with the notion of orientation.

3 Orientation

Consider two orthonormal bases for a Euclidean space $\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{z}$ and $\tilde{\boldsymbol{x}}, \tilde{\boldsymbol{y}}, \tilde{\boldsymbol{z}}$. An *orientation* is an expression of the axis directions of one basis in terms of the other. To illustrate this notion, consider an aircraft at time t_0 aligned with one basis: the center of mass of the airplane is at the vector space origin, the fuselage points in direction $\boldsymbol{x}$, the starboard wing points in direction $\boldsymbol{y}$, and (to complete a right handed system) $\boldsymbol{z}$ points down with respect to the aircraft (see **Figure 7**). At some later time t_1 subject the airplane to a roll, pitch, and yaw. We subtract the vector that represents the displacement of the center of mass from time t_0 to time t_1 and the new directions for the fuselage, starboard wing, and relative down define the $\tilde{\boldsymbol{x}}, \tilde{\boldsymbol{y}}, \tilde{\boldsymbol{z}}$ directions. The two vector spaces spanned by bases $\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{z}$ and $\tilde{\boldsymbol{x}}, \tilde{\boldsymbol{y}}, \tilde{\boldsymbol{z}}$ share the same origin and are thus the same vector space. The only difference is that $\tilde{\boldsymbol{x}}, \tilde{\boldsymbol{y}}, \tilde{\boldsymbol{z}}$ has a different orientation with respect to $\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{z}$. Orientation is also called *attitude* in some contexts.

3.1 Orientation and Rotation

Let $\boldsymbol{r}$ and $\tilde{\boldsymbol{r}}$ respectively denote the coordinate representation of a point with respect to each basis.

With respect to basis $\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{z}$, $\boldsymbol{r} = (r_1, r_2, r_3)^{\mathrm{T}}$, where $\boldsymbol{r} = r_1\boldsymbol{x} + r_2\boldsymbol{y} + r_3\boldsymbol{z}$.

With respect to basis $\tilde{\boldsymbol{x}}, \tilde{\boldsymbol{y}}, \tilde{\boldsymbol{z}}$, $\tilde{\boldsymbol{r}} = (\tilde{r}_1, \tilde{r}_2, \tilde{r}_3)^{\mathrm{T}}$, where $\tilde{\boldsymbol{r}} = \tilde{r}_1\tilde{\boldsymbol{x}} + \tilde{r}_2\tilde{\boldsymbol{y}} + \tilde{r}_3\tilde{\boldsymbol{z}}$.

The transformation of a coordinate $(r_1, r_2, r_3) \mapsto (\tilde{r}_1, \tilde{r}_2, \tilde{r}_3)$ is a linear transformation and can thus be realized as a matrix multiplication:

$$\begin{pmatrix} \tilde{r}_1 \\ \tilde{r}_2 \\ \tilde{r}_3 \end{pmatrix} = \boldsymbol{Q} \begin{pmatrix} r_1 \\ r_2 \\ r_3 \end{pmatrix} = \begin{pmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{pmatrix} \begin{pmatrix} r_1 \\ r_2 \\ r_3 \end{pmatrix}$$

where:

$$\begin{aligned} a_{11} &= \boldsymbol{x} \bullet \tilde{\boldsymbol{x}}, & a_{12} &= \boldsymbol{y} \bullet \tilde{\boldsymbol{x}}, & a_{13} &= \boldsymbol{z} \bullet \tilde{\boldsymbol{x}} \\ a_{21} &= \boldsymbol{x} \bullet \tilde{\boldsymbol{y}}, & a_{22} &= \boldsymbol{y} \bullet \tilde{\boldsymbol{y}}, & a_{23} &= \boldsymbol{z} \bullet \tilde{\boldsymbol{y}} \\ a_{31} &= \boldsymbol{x} \bullet \tilde{\boldsymbol{z}}, & a_{32} &= \boldsymbol{y} \bullet \tilde{\boldsymbol{z}}, & a_{33} &= \boldsymbol{z} \bullet \tilde{\boldsymbol{z}} \end{aligned} \tag{1.1}$$

Since the basis vectors are unit vectors, the dot products are the cosine of the angle between the two vectors. For this reason the matrix is the called *direction cosine matrix*. Note that the columns of the matrix are the $\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{z}$ basis vectors in $\tilde{\boldsymbol{x}}, \tilde{\boldsymbol{y}}, \tilde{\boldsymbol{z}}$ coordinate representation while the rows (or columns of the transpose matrix) are the $\tilde{\boldsymbol{x}}, \tilde{\boldsymbol{y}}, \tilde{\boldsymbol{z}}$ basis vectors in $\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{z}$ coordinate representation.

Euler's rotation theorem states that this linear transformation is a rotation operation. In particular, the matrix has a unit eigenvector $\boldsymbol{n}$ and three eigenvalues: 1, $e^{+i\theta}$, $e^{-i\theta}$. The line spanned by $\boldsymbol{n}$ is fixed under the transformation and represents the axis of rotation. The angle of rotation is given by θ.

Euler's rotation theorem thus shows that orientation and rotation are just two ways of viewing the same transformation. These two ways are closely related, but are <u>not</u> equivalent. Consider Figure 5. On the left side, the point $\boldsymbol{r}$ is rotated by angle θ about the z-axis (which points directly toward the reader) to a new position $\boldsymbol{r}'$.

The coordinates of these two points, $\boldsymbol{r} = (r_1, r_2, r_3)^{\mathrm{T}}$, and $\boldsymbol{r}' = (r_1', r_2', r_3')^{\mathrm{T}}$ are related by the following matrix.

$$\begin{pmatrix} r_1' \\ r_2' \\ r_3' \end{pmatrix} = \boldsymbol{R}_z(\theta) \begin{pmatrix} r_1 \\ r_2 \\ r_3 \end{pmatrix} = \begin{pmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} r_1 \\ r_2 \\ r_3 \end{pmatrix}.$$

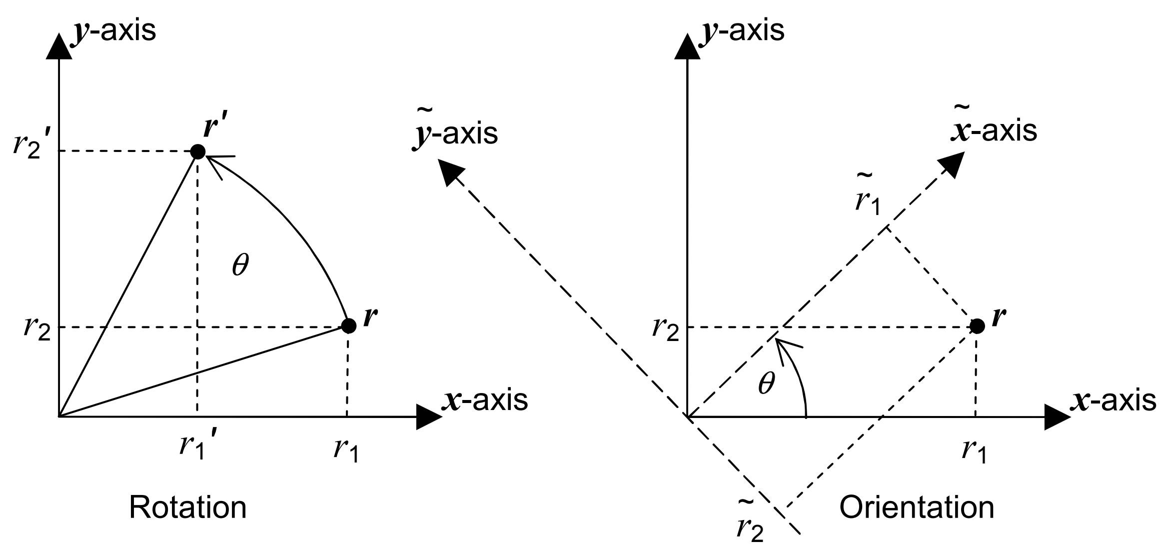

Figure 5 – Rotation and orientation

The right side of the figure shows a second basis whose orientation with respect to the first basis is a rotation by angle θ about the z-axis. The coordinates of the single point $\boldsymbol{r}$

are related by the following direction cosine matrix.

$$\begin{pmatrix} \tilde{r}_1 \\ \tilde{r}_2 \\ \tilde{r}_3 \end{pmatrix} = \boldsymbol{\Omega}_{\boldsymbol{z}} \begin{pmatrix} r_1 \\ r_2 \\ r_3 \end{pmatrix} = \begin{pmatrix} \cos\theta & \sin\theta & 0 \\ -\sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} r_1 \\ r_2 \\ r_3 \end{pmatrix}.$$

Notice that matrices corresponding to the left and right figures are not the same: $\boldsymbol{R}_{\boldsymbol{z}}(\theta) = \boldsymbol{\Omega}_{\boldsymbol{z}}^{\mathrm{T}}$, and $\boldsymbol{\Omega}_{\boldsymbol{z}} \boldsymbol{R}_{\boldsymbol{z}}(\theta) = \boldsymbol{I}_{3\times3}$. So while both cases, the rotation of a point, and the orientation of a coordinate system, involve the same axis of rotation and the same angle of rotation, the corresponding linear operations are, in fact, the inverses of each other. We shall call an operator that performs a rotation, such as the operator on the left side of Figure 1, a *rotation operator* and an operator that changes coordinate system directions, such as on the right side of the Figure, an *orientation operator*.

Note that to transform a coordinate from the $\tilde{\boldsymbol{x}}, \tilde{\boldsymbol{y}}, \tilde{\boldsymbol{z}}$ coordinate system back to the original coordinate system, the rotation matrix may be used:

$$\begin{pmatrix} r_1 \\ r_2 \\ r_3 \end{pmatrix} = \boldsymbol{\Omega}_{\boldsymbol{z}}^{-1} \begin{pmatrix} \tilde{r}_1 \\ \tilde{r}_2 \\ \tilde{r}_3 \end{pmatrix} = \boldsymbol{R}_{\boldsymbol{z}}(\theta) \begin{pmatrix} \tilde{r}_1 \\ \tilde{r}_2 \\ \tilde{r}_3 \end{pmatrix}$$

It follows that a representation of a rotation will depend on its intended use or interpretation. This document will address three primary use cases:

<u>Primary use case 1.</u> This primary use case concerns rigid body dynamics. Rigid body dynamics characterize the motion of a rigid body by translation and rotation. Of particular concern in this document are the characterizations of instantaneous rotational kinematics – rotational velocity and rotational acceleration, and rotational dynamics – torque and inertia.

<u>Primary use case 2.</u> This primary use case concerns the descriptions of point positions in one coordinate system with respect to another other coordinate system with a different orientation. A sub-case concerns position descriptions between a “space-fixed” or inertial coordinate system and “body-fixed” coordinate system attached to a rigid body that is either static or moving in time.

<u>Primary use case 3.</u> This primary use case combines the first two. Of particular concern is representing rigid body dynamics characterizations computed in one coordinate system in terms of the second coordinate system. The coordinate systems may both be space-fixed, or one may be moving with respect to the other.

3.2 Representing Rotations

Rigid body motion exhibits six degrees of freedom - three degrees of freedom for translation and three degrees of freedom for rotation. This means that, in principle, a rotation operation on 3D Euclidean space can be specified by three scalar numbers. That is indeed the case with Euler angles (see below). However, other less compact specifications are used because they are more amenable to some computations such as performing a rotation operation on a vector, composing rotations, interpolating rotations, and other operations, and/or because they can be measured or modeled directly. Of the various representation methods in prevalent use, each presents various tradeoffs with respect to storage size, and computational complexity, speed, and error control. Thus the best representation is dependent on the requirements and computational environment of a user application. For this reason, different representations are in use and interoperability becomes an issue. This issue is compounded by the non-standard meaning of terms in prevalent uses. To support interoperability, this document will define terms and various methods and present algorithms for both key operations and inter-conversions between the representation methods. The first rotation representation method is the axis-angle form.

3.2.1 Axis-angle, vector with right handed rotation

The *axis-angle* representation of a rotation, $(\boldsymbol{n}, \theta)$, consists of a unit vector $\boldsymbol{n}$ $(\|\boldsymbol{n}\| = 1)$ and a rotation angle θ. This represents a rotation through angle θ about the axis spanned by $\boldsymbol{n}$. The rotation direction is determined by the *right hand rule*: conceptually, if the right hand holds the vector $\boldsymbol{n}$ with thumb pointing in the direction of the vector, the fingers point in the direction of increasing θ.

This representation uses four scalar parameters $\boldsymbol{n} = (n_1, n_2, n_3)$ and θ. The constraint $\|\boldsymbol{n}\| = 1$ reduces the degrees of freedom down to three degrees of freedom.

3.2.1.1 Rodrigues' rotation formula

The rotation of a vector $\boldsymbol{r}$ to a rotated vector $\boldsymbol{r}'$ in terms of $(\boldsymbol{n}, \theta)$ is given by Rodrigues' rotation formula (see Appendix B):

$$\boldsymbol{r}' = \cos(\theta)\boldsymbol{r} + (1 - \cos(\theta))(\boldsymbol{r} \bullet \boldsymbol{n})\boldsymbol{n} + \sin(\theta)\boldsymbol{n} \times \boldsymbol{r} \tag{1.2}$$

The terms may be rearranged to the alternate form:

$$\boldsymbol{r}' = \boldsymbol{r} + (1 - \cos(\theta))\boldsymbol{n} \times (\boldsymbol{n} \times \boldsymbol{r}) + \sin(\theta)\boldsymbol{n} \times \boldsymbol{r} \tag{1.3}$$

The matrix form of this formula is:

$$\boldsymbol{r}' = \boldsymbol{R}\,\boldsymbol{r}$$

where:

$$\boldsymbol{R} = \left[\boldsymbol{I}_{3\times3} + \sin(\theta)\boldsymbol{S}_{\boldsymbol{n}} + (1 - \cos(\theta))\boldsymbol{S}_{\boldsymbol{n}}^2\right] \tag{1.4}$$

or, alternatively (see Appendix B):

$$\boldsymbol{R} = \left[\cos(\theta)\boldsymbol{I}_{3\times3} + (1 - \cos(\theta))\boldsymbol{n} \otimes \boldsymbol{n} + \sin(\theta)\boldsymbol{S}_{\boldsymbol{n}}\right] \tag{1.5}$$

and

$$S_n = \begin{pmatrix} 0 & -n_3 & n_2 \\ n_3 & 0 & -n_1 \\ -n_2 & n_1 & 0 \end{pmatrix}$$

is the skew-symmetric matrix associated with $\boldsymbol{n}$.

3.2.2 Principal rotations

For a given 3 dimensional Euclidean space, the three vectors: $\boldsymbol{x} = (1,0,0)^{\mathsf{T}}$, $\boldsymbol{y} = (0,1,0)^{\mathsf{T}}$, and $\boldsymbol{z} = (0,0,1)^{\mathsf{T}}$ form an orthonormal basis for the vector space. As an axis of rotation, each is called a *principal axis*[1] of rotation. A rotation about a principal axis is called a *principal rotation*. Some authors refer to these rotations as *elementary Rotations*. The vector space operators: $\boldsymbol{R_x}(\alpha)$, $\boldsymbol{R_y}(\beta)$, and $\boldsymbol{R_z}(\gamma)$ will denote the principal rotations through the respective angles α, β, and γ.

3.2.3 Rotation Matrix

A rotation operation on vector space is a linear operator, thus for a given basis it has a matrix representation (see 3.1). If $\boldsymbol{M}$ is a rotation matrix, it satisfies these properties:

$$\begin{aligned} &\det(\boldsymbol{M}) = 1 \\ &\boldsymbol{M}\boldsymbol{M}^{\mathsf{T}} = \boldsymbol{M}^{\mathsf{T}}\boldsymbol{M} = \boldsymbol{I}_{3\times 3} \end{aligned} \tag{1.6}$$

This property implies that $\boldsymbol{M}^{-1} = \boldsymbol{M}^{\mathsf{T}}$. Matrices satisfying these properties form a group with respect to matrix multiplication. This group is known as the special orthogonal group of degree 3, SO(3). In particular, the product of any two rotation matrices is itself a rotation matrix. (Note: Matrix multiplication is generally not commutative).

Performing a rotation operation may be realized by simple matrix multiplication: $\boldsymbol{r}' = \boldsymbol{M}\boldsymbol{r}$. The inverse operation is $\boldsymbol{r} = \boldsymbol{M}^{\mathsf{T}}\boldsymbol{r}'$. While this is computationally convenient, the matrix representation does not lend itself well to intuitive visualization of the corresponding rotation. For this and other reasons, it is important to able to factor a given rotation matrix into principal rotations. The matrix forms of the principal rotations are:

[1] This term should not be confused with the moment of inertia principal axes.

$$\boldsymbol{R_x}(\alpha) = \boldsymbol{\Omega}_{\boldsymbol{x}}^{\mathrm{T}}(\alpha) = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos\alpha & -\sin\alpha \\ 0 & \sin\alpha & \cos\alpha \end{pmatrix},$$

$$\boldsymbol{R_y}(\beta) = \boldsymbol{\Omega}_{\boldsymbol{y}}^{\mathrm{T}}(\beta) = \begin{pmatrix} \cos\beta & 0 & \sin\beta \\ 0 & 1 & 0 \\ -\sin\beta & 0 & \cos\beta \end{pmatrix}, \text{ and}$$

$$\boldsymbol{R_z}(\gamma) = \boldsymbol{\Omega}_{\boldsymbol{z}}^{\mathrm{T}}(\gamma) = \begin{pmatrix} \cos\gamma & -\sin\gamma & 0 \\ \sin\gamma & \cos\gamma & 0 \\ 0 & 0 & 1 \end{pmatrix}. \tag{1.7}$$

NOTE: We are using $\boldsymbol{R}$ to denote the rotation operation that moves a point by rotation, and $\boldsymbol{\Omega}$ to denote the orientation operation that transforms a coordinate in a coordinate system into a coordinate in a coordinate system that is rotated with respect to the first coordinate system.

To factor a rotation matrix $\boldsymbol{R} = \begin{pmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{pmatrix}$ in to a sequence of principal rotations of the form[2] $\boldsymbol{R_z}(\gamma)\boldsymbol{R_y}(\beta)\boldsymbol{R_x}(\alpha)$, we multiply the corresponding matrices to obtain:

$$\boldsymbol{R_z}(\gamma)\boldsymbol{R_y}(\beta)\boldsymbol{R_x}(\alpha) =$$
$$\begin{pmatrix} \cos\gamma\cos\beta & \cos\gamma\sin\beta\sin\alpha - \sin\gamma\cos\beta & \cos\gamma\sin\beta\cos\alpha + \sin\gamma\sin\alpha \\ \sin\gamma\cos\beta & \sin\gamma\sin\beta\sin\alpha + \cos\gamma\cos\alpha & \sin\gamma\sin\beta\cos\alpha - \cos\gamma\sin\alpha \\ -\sin\beta & \cos\beta\sin\alpha & \cos\beta\cos\alpha \end{pmatrix} \tag{1.8}$$

Matching the elements of this matrix to those of $\boldsymbol{R}$ we find that $a_{31} = -\sin\beta$, $a_{32}/a_{33} = \tan\alpha$, and $a_{21}/a_{11} = \tan\gamma$. These equations lead to the solutions in Table 1 based on the value of a_{31}.

Table 1–Principal rotation factors for *z*-*y*-*x* rotation

Case	**Principal rotation factors for $\boldsymbol{R_z}(\gamma)\boldsymbol{R_y}(\beta)\boldsymbol{R_x}(\alpha)$**

[2] Note that the order of applying each of principal rotations is right-to-left so that the $\boldsymbol{R_x}(\alpha)$ rotation is performed first.

Case	Principal rotation factors for $\boldsymbol{R_z}(\gamma)\boldsymbol{R_y}(\beta)\boldsymbol{R_x}(\alpha)$		
$a_{31} \neq \pm 1$	$\beta = \text{arcsin}(-a_{31})$ $-\pi/2 < \beta < \pi/2$	$\alpha = \text{atan2}(a_{32}, a_{33})$	$\gamma = \text{atan2}(a_{21}, a_{11})$
	$\beta = \text{arcsin}(-a_{31})$ $\pi/2 < \beta < 3\pi/2$	$\alpha = \text{atan2}(-a_{32}, -a_{33})$	$\gamma = \text{atan2}(-a_{21}, -a_{11})$
$a_{31} = -1$	$\beta = \pi/2$	$\alpha = \text{atan2}(a_{12}, a_{13}) + \gamma$	any value of γ
$a_{31} = +1$	$\beta = -\pi/2$	$\alpha = \text{atan2}(a_{12}, a_{13}) - \gamma$	any value of γ

In the case $a_{31} \neq \pm 1$ there are two valid solution sets depending on the quadrants selected for arcsine values. In the case $a_{31} = -1$, using the trigonometric identities for the difference of angles and substituting $\sin\beta = 1$ and $\cos\beta = 0$, the matrix reduces to :

$$\boldsymbol{R_z}(\gamma)\boldsymbol{R_y}\left(\frac{\pi}{2}\right)\boldsymbol{R_x}(\alpha) = \begin{pmatrix} 0 & \sin(\alpha-\gamma) & \cos(\alpha-\gamma) \\ 0 & \cos(\alpha-\gamma) & -\sin(\alpha-\gamma) \\ -1 & 0 & 0 \end{pmatrix}.$$

This shows that only the difference of the other two angles is determined as $\alpha - \gamma = \text{atan2}(a_{12}, a_{13})$. Therefore all values are valid for γ if we set $\alpha = \text{atan2}(a_{12}, a_{13}) + \gamma$. The case $a_{31} = +1$ is similar to the previous case with only the sum of angles is determined as $\alpha + \gamma = \text{atan2}(a_{12}, a_{13})$.

The non-determinacy in the last two cases is related to the phenomenon of gimbal lock (see below).

To factor an <u>orientation</u> matrix $\boldsymbol{\Omega} = \begin{pmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{pmatrix}$ into a sequence of principal rotations of the form $\boldsymbol{\Omega_x}(\alpha)\boldsymbol{\Omega_y}(\beta)\boldsymbol{\Omega_z}(\gamma)$, we multiply the corresponding matrices to obtain:

$$\boldsymbol{\Omega_x}(\alpha)\,\boldsymbol{\Omega_y}(\beta)\,\boldsymbol{\Omega_z}(\gamma) =$$
$$\begin{pmatrix} \cos\gamma\cos\beta & \sin\gamma\cos\beta & -\sin\beta \\ \cos\gamma\sin\beta\sin\alpha - \sin\gamma\cos\alpha & \sin\gamma\sin\beta\sin\alpha + \cos\gamma\cos\alpha & \cos\beta\sin\alpha \\ \cos\gamma\sin\beta\cos\alpha + \sin\gamma\sin\alpha & \sin\gamma\sin\beta\cos\alpha - \cos\gamma\sin\alpha & \cos\beta\cos\alpha \end{pmatrix} \tag{1.9}$$

Note that this matrix is just the transpose of $\boldsymbol{R_z}(\gamma)\,\boldsymbol{R_y}(\beta)\,\boldsymbol{R_x}(\alpha)$ so that solutions use transposed elements as shown in Table 2.

Table 2 – Principal orientation factors for *x-y-z* orientation

Case	Principal rotation factors for $\boldsymbol{\Omega_x}(\alpha)\,\boldsymbol{\Omega_y}(\beta)\,\boldsymbol{\Omega_z}(\gamma)$		
$a_{13} \neq \pm 1$	$\beta = \arcsin(-a_{13})$ $-\pi/2 < \beta < \pi/2$	$\alpha = \text{atan2}(a_{23}, a_{33})$	$\gamma = \text{atan2}(a_{12}, a_{11})$
	$\beta = \arcsin(-a_{13})$ $\pi/2 < \beta < 3\pi/2$	$\alpha = \text{atan2}(-a_{23}, -a_{33})$	$\gamma = \text{atan2}(-a_{12}, -a_{11})$
$a_{13} = -1$	$\beta = \pi/2$	$\alpha = \text{atan2}(a_{21}, a_{31}) + \gamma$	any value of γ
$a_{13} = +1$	$\beta = -\pi/2$	$\alpha = \text{atan2}(a_{21}, a_{31}) - \gamma$	any value of γ

As can be seen in Table 1 and Table 2, the three angle sequence corresponding to a given rotation or orientation operator is not unique. Two such sequences, $(\alpha_1, \beta_1, \gamma_1)$ and $(\alpha_2, \beta_2, \gamma_2)$ specify the same operator if they satisfy one the criteria of the next Table.

Table 3 – Equivalence of *x-y-z* principal rotation sequences

Case	Criteria for the equivalence of *x-y-z* principal rotation sequences: $(\alpha_1, \beta_1, \gamma_1)$ and $(\alpha_2, \beta_2, \gamma_2)$
$\beta_1 = \beta_2$	$\alpha_1 = \alpha_2,\ \gamma_1 = \gamma_2$ $\quad \left[\beta_1 \neq \pm\frac{\pi}{2} \neq \beta_2\right]$

Case	**Criteria for the equivalence of *x-y-z* principal rotation sequences:** $(\alpha_1, \beta_1, \gamma_1)$ **and** $(\alpha_2, \beta_2, \gamma_2)$
$\beta_1 + \beta_2 = \pi$	$\lvert\alpha_2 - \alpha_1\rvert = \pi, \ \lvert\gamma_2 - \gamma_1\rvert = \pi \qquad \left[\beta_1 \neq \frac{\pi}{2} \neq \beta_2\right]$
$\beta_1 = \beta_2 = \frac{\pi}{2}$	$\alpha_1 - \gamma_1 = \alpha_2 - \gamma_2$
$\beta_1 = \beta_2 = -\frac{\pi}{2}$	$\alpha_1 + \gamma_1 = \alpha_2 + \gamma_2$

Factorizations using other axis orderings may be obtained in similar fashion. There are twelve non-trivial orderings. The case $\boldsymbol{\Omega}_z(\gamma)\boldsymbol{\Omega}_x(\beta)\boldsymbol{\Omega}_z(\alpha)$ is used below. The orientation matrix is then in the form:

$$\boldsymbol{\Omega}_z(\gamma)\boldsymbol{\Omega}_x(\beta)\boldsymbol{\Omega}_z(\alpha) = \begin{pmatrix} \cos\alpha\cos\gamma - \cos\beta\sin\alpha\sin\gamma & \cos\beta\cos\alpha\sin\gamma + \sin\alpha\cos\gamma & \sin\beta\sin\gamma \\ -\cos\alpha\sin\gamma - \cos\beta\sin\alpha\cos\gamma & \cos\beta\cos\alpha\cos\gamma - \sin\alpha\sin\gamma & \sin\beta\cos\gamma \\ \sin\beta\sin\alpha & -\sin\beta\cos\alpha & \cos\beta \end{pmatrix} \tag{1.10}$$

The Principal rotation factors for this ordering are shown in Table 4.

Table 4–Principal rotation factors for *z-x-z* orientation

Case	Principal rotation factors for $\boldsymbol{\Omega}_z(\gamma)\boldsymbol{\Omega}_x(\beta)\boldsymbol{\Omega}_z(\alpha)$		
$a_{33} \neq \pm 1$	$\beta = \text{arccos}(a_{33})$ $0 < \beta < \pi$	$\alpha = \text{atan2}(a_{31}, -a_{32})$	$\gamma = \text{atan2}(a_{13}, a_{23})$
	$\beta = \text{arccos}(a_{33})$ $\pi < \beta < 2\pi$	$\alpha = \text{atan2}(-a_{31}, a_{32})$	$\gamma = \text{atan2}(-a_{13}, -a_{23})$

$a_{33} = -1$	$\beta = \pi$	$\alpha = \text{atan2}(a_{12}, a_{11}) + \gamma$	any value of γ
$a_{33} = +1$	$\beta = 0$	$\alpha = \text{atan2}(a_{12}, a_{11}) - \gamma$	any value of γ

Two sequences, $(\alpha_1, \beta_1, \gamma_1)$ and $(\alpha_2, \beta_2, \gamma_2)$ of z-x-z principal rotations specify the same operator if they satisfy one the criteria of the next Table.

Table 5 – Equivalence of z-y-z principal rotation sequences

Case	Criteria for the equivalence of z-y-z principal rotation sequences: $(\alpha_1, \beta_1, \gamma_1)$ and $(\alpha_2, \beta_2, \gamma_2)$
$\beta_1 = \beta_2$	$\alpha_1 = \alpha_2,\ \gamma_1 = \gamma_2 \qquad [\beta_1, \beta_2 \neq 0 \text{ or } \pi]$
$\beta_1 + \beta_2 = 2\pi$	$\lvert\alpha_2 - \alpha_1\rvert = \pi,\ \lvert\gamma_2 - \gamma_1\rvert = \pi \qquad [\beta_1, \beta_2 \neq 0 \text{ or } \pi]$
$\beta_1 = \beta_2 = \pi$	$\alpha_1 - \gamma_1 = \alpha_2 - \gamma_2$
$\beta_1 = \beta_2 = 0$	$\alpha_1 + \gamma_1 = \alpha_2 + \gamma_2$

3.2.4 Euler angles

Euler angles are a specification of a rotation obtained by applying three consecutive principal rotations. There are twelve distinct ways to select a sequence pf three principal axes and apply the principal rotations (24 if left-handed axes are considered). Each such ordered selection is an *Euler angle convention*. There is little agreement among authors in naming these conventions.

The three principal rotations may be either rotations about the original axes, or about the successively rotated axes. Given a rotation, let $\tilde{\boldsymbol{x}}, \tilde{\boldsymbol{y}}, \tilde{\boldsymbol{z}}$ be the principal axes after the rotation is applied to the $\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{z}$ axes. To distinguish between these two coordinate bases, coordinates with respect to the original basis $\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{z}$ will be called *space-fixed* or static coordinates and those with respect to the moving $\tilde{\boldsymbol{x}}, \tilde{\boldsymbol{y}}, \tilde{\boldsymbol{z}}$ axes will be called *body-fixed* or rotating coordinates. It is useful to think of the $\tilde{\boldsymbol{x}}, \tilde{\boldsymbol{y}}, \tilde{\boldsymbol{z}}$ as attached to a rigid body that will be rotated. We shall assume that the $\boldsymbol{xy}$-plane and $\tilde{\boldsymbol{x}}\tilde{\boldsymbol{y}}$-plane intersect in a line. This line is called the *line of nodes*.

The *Euler angles* in the z-x-z convention are the three angles defined as follows:

> α is the angle between the x-axis and the line of nodes,
> β is the angle between z-axis and the $\tilde{z}$-axis, and
> γ is the angle between the $\tilde{x}$-axis and the line of nodes.

In some contexts α is called the *spin* angle, β is called the *nutation* angle, and γ is called the *precession* angle. Many authors use the symbols ϕ, θ, and ψ for these angles, but disagree on the identifying ordering.

These three angles specify a rotation as consecutive principal rotations using the z–axis, the x–axis and z–axis again. There are two equivalent specifications, space-fixed (or static) and body-fixed or rotating.

In the space-fixed (or static) specification, all the principal rotations are about the fixed-space principal axes z and x. The first principal rotation is about the z-axis through angle α, followed by the x-axis through angle β, followed by the z-axis again through angle γ. The combined rotation is $\boldsymbol{R}_z(\gamma)\boldsymbol{R}_x(\beta)\boldsymbol{R}_z(\alpha)$.

The assumption that the xy-plane and $\tilde{x}\tilde{y}$-plane intersect in a line is met when $\beta \neq 0$ and $\beta \neq \pi$. If $\beta = 0$ or $\beta = \pi$, the consecutive rotations collapse down to a single principal rotation (compare with Table 4):

$$\begin{aligned}\beta = 0: &\quad \boldsymbol{R}_z(\gamma)\boldsymbol{R}_x(0)\boldsymbol{R}_z(\alpha) = \boldsymbol{R}_z(\gamma)\boldsymbol{R}_z(\ \ \alpha) = \boldsymbol{R}_z(\gamma+\alpha)\\ \beta = \pi: &\quad \boldsymbol{R}_z(\gamma)\boldsymbol{R}_x(\pi)\boldsymbol{R}_z(\alpha) = \boldsymbol{R}_z(\gamma)\boldsymbol{R}_z(-\alpha) = \boldsymbol{R}_z(\gamma-\alpha)\end{aligned} \tag{1.11}$$

In the body-fixed (or rotating) specification, all the principal rotations are about the body-space principal axes $\tilde{z}$ and $\tilde{x}$. Before any rotation is applied, the space-fixed and body-fixed bases coincide. The first principal rotation is about the z-axis through angle γ. This rotates axes $\tilde{x}$ and $\tilde{y}$ to the intermediate orientations x' and y' (in this intermediate position, the x'-axis lies on the line of nodes). The second rotation is about the intermediate x'-axis through angle β. This second rotation moves the y' and z axes to intermediate orientations z'' and y''. The third rotation is about z''-axis through α which moves axes x' and y'' to their final orientations x''' and y''' The combined rotation is $\boldsymbol{R}_{z''}(\alpha)\boldsymbol{R}_{x'}(\beta)\boldsymbol{R}_z(\gamma)$. The final body-fixed axis orientations are $\tilde{x} = x'''$, $\tilde{y} = y'''$, $\tilde{z} = z''$. The sequence of body-fixed rotations is illustrated in Figure 6.

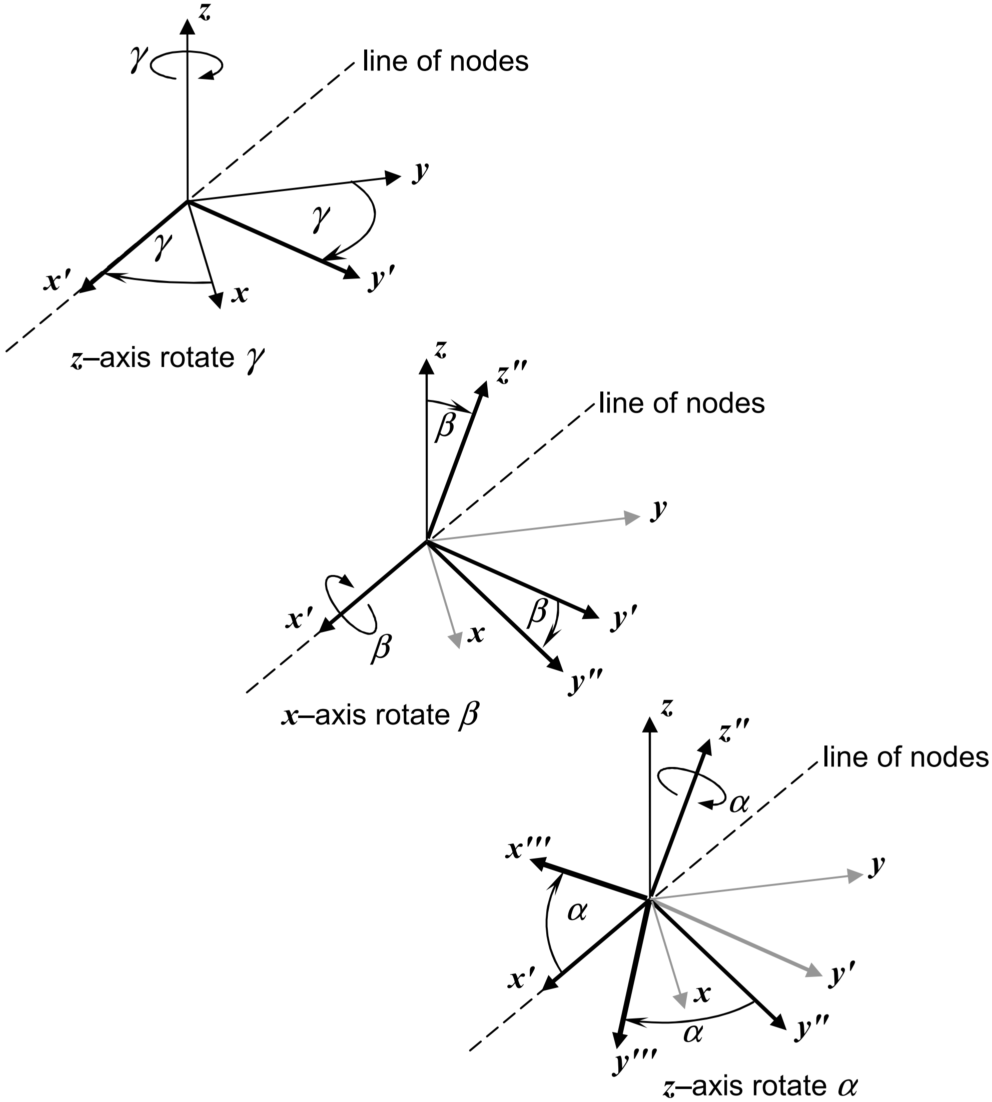

Figure 6 — Euler z-x-z rotation sequence

Observe that order of the three rotation angles is reversed between the space- and body-fixed cases:

$$\begin{aligned} &\boldsymbol{R}_z(\gamma)\boldsymbol{R}_x(\beta)\boldsymbol{R}_z(\alpha) \quad \text{space-fixed} \\ &\boldsymbol{R}_{z''}(\alpha)\boldsymbol{R}_{x'}(\beta)\boldsymbol{R}_z(\gamma) \quad \text{body-fixed} \end{aligned} \tag{1.12}$$

To show that both expressions produce the same rotation, note that when $\boldsymbol{x'}$ is at its intermediate position on the line of nodes, the second rotation $\boldsymbol{R}_{x'}(\beta)$ is equivalent to first rotating the line of nodes to the $\boldsymbol{x}$-axis using principal rotation $\boldsymbol{R}_z(-\gamma)$, rotating about the $\boldsymbol{x}$-axis (which is the line of nodes at this point) with $\boldsymbol{R}_x(\beta)$ and finally rotating the line of nodes back to its original position with $\boldsymbol{R_z}(\gamma)$. In effect, $\boldsymbol{R_{x'}}(\beta)=\boldsymbol{R_z}(\gamma)\boldsymbol{R_x}(\beta)\boldsymbol{R_z}(-\gamma)$. Similarly, $\boldsymbol{R_{z''}}(\alpha)=\boldsymbol{R_{x'}}(\beta)\boldsymbol{R_z}(\alpha)\boldsymbol{R_{x'}}(-\beta)$. Noting that two rotations about the same axis commute and substituting these expressions in the body-fixed formulation gives:

$$\begin{aligned}
\boldsymbol{R_{z''}}(\alpha)\boldsymbol{R_{x'}}(\beta)\boldsymbol{R_z}(\gamma) &= \left[\boldsymbol{R_{x'}}(\beta)\boldsymbol{R_z}(\alpha)\boldsymbol{R_{x'}}(-\beta)\right]\boldsymbol{R_{x'}}(\beta)\boldsymbol{R_z}(\gamma) \\
&= \left[\boldsymbol{R_{x'}}(\beta)\boldsymbol{R_z}(\alpha)\right]\boldsymbol{R_z}(\gamma) \\
&= \left[\left\{\boldsymbol{R_z}(\gamma)\boldsymbol{R_x}(\beta)\boldsymbol{R_z}(-\gamma)\right\}\boldsymbol{R_z}(\alpha)\right]\boldsymbol{R_z}(\gamma) \\
&= \boldsymbol{R_z}(\gamma)\boldsymbol{R_x}(\beta)\boldsymbol{R_z}(\alpha)\boldsymbol{R_z}(-\gamma)\boldsymbol{R_z}(\gamma) \\
&= \boldsymbol{R_z}(\gamma)\boldsymbol{R_x}(\beta)\boldsymbol{R_z}(\alpha)
\end{aligned}$$

This result:

$$\boldsymbol{R_{z''}}(\alpha)\boldsymbol{R_{x'}}(\beta)\boldsymbol{R_z}(\gamma)=\boldsymbol{R_z}(\gamma)\boldsymbol{R_x}(\beta)\boldsymbol{R_z}(\alpha) \tag{1-13}$$

shows that the space-fixed and body-fixed formulations produce the same rotation. Both formulations are important. The matrix formulations of the principal rotations (Equation (1.7)) are expressed with respect to the static frame. However, a gyroscope attached to the rotated body would read out the angles with respect to the rotating frame.

The orientation of the $\tilde{x},\tilde{y},\tilde{z}$ axes with respect to the x, y, z is inverse of the rotation:

$$\boldsymbol{\Omega_z}(\alpha)\boldsymbol{\Omega_x}(\beta)\boldsymbol{\Omega_z}(\gamma) \tag{1.14}$$

3.2.4.1 Other conventions

There are numerous conventions for Euler angles in use and many are named inconsistently. (Note that some authors use a left-handed coordinate system. All coordinate systems in this document are right-handed). The convention defined above using axes $z\text{–}x\text{–}z$ (also known as the 3-1-3 convention) is often called the x-convention. Replacing x with y gives the so called y-convention ($z\text{–}y\text{–}z$ or 3-2-3). Quantum physics treatments prefer the y-convention, but $x\text{–}y\text{–}x$ (or 1-2-1) is also called the y-convention by some authors. The convention using $x\text{–}y\text{–}z$ (or 1-2-3) is defined below.

3.2.4.2 The $x\text{–}y\text{–}z$ convention (Tait-Bryan angles)

This convention is used to specify orientation in DIS packets as specified in the IEEE 1278.1-1995 standard. Euler angles in this convention are variously called *Tait-Bryan angles*, *Cardano angles*, or *nautical angles*. In this convention the line of nodes is the intersection of the xy-plane and $\tilde{y}\tilde{z}$-plane. The Euler angles in the $x\text{–}y\text{–}z$ convention are defined as follows:

ψ is the angle between the $\boldsymbol{y}$-axis and the line of nodes,
θ is the angle between $\boldsymbol{z}$-axis and the $\tilde{\boldsymbol{y}}\tilde{\boldsymbol{z}}$-plane, and
ϕ is the angle between the $\tilde{\boldsymbol{y}}$-axis and the line of nodes.

The rotation is specified as principal rotations about the body-space principal axes. The first rotation is by angle ψ about the $\tilde{\boldsymbol{z}}$-axis. The second is by angle θ about the $\tilde{\boldsymbol{y}}$-axis. The third is by angle ϕ about the $\tilde{\boldsymbol{x}}$-axis. The combined rotation is

$$\boldsymbol{R}_{\tilde{\boldsymbol{x}}}(\phi)\boldsymbol{R}_{\tilde{\boldsymbol{y}}}(\theta)\boldsymbol{R}_{\tilde{\boldsymbol{z}}}(\psi). \tag{1.15}$$

The equivalent space-fixed (or static) specification is:

$$\boldsymbol{R}_{\boldsymbol{z}}(\psi)\boldsymbol{R}_{\boldsymbol{y}}(\theta)\boldsymbol{R}_{\boldsymbol{x}}(\phi). \tag{1.16}$$

The various names given to these angle symbols include:

ϕ roll or bank or tilt,
θ pitch or elevation, and
ψ yaw or heading or azimuth.

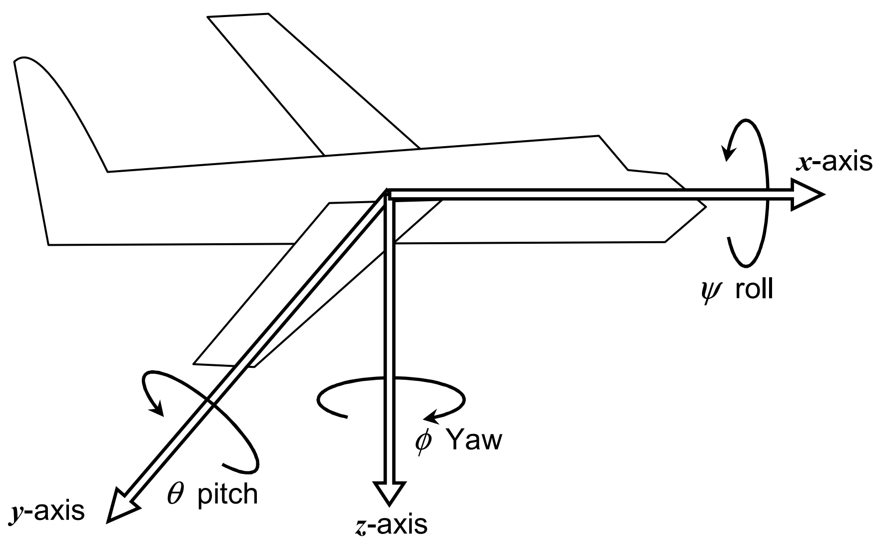

Figure 7 - Tait-Bryan angles

When the fixed body is an aircraft, the common practice is to choose the center of mass as the coordinate system origin with the $\boldsymbol{x}$-axis pointing forward, the $\boldsymbol{y}$-axis pointing starboard, and the z-axis pointing down (to complete a right handed system[3]). This is also the "entity coordinate system" defined in the IEEE 1278.1-1995 standard. In these cases the Euler angles are called Tait-Bryan angles and the names *roll*, *pitch*, and *yaw* are specifically used.

In the IEEE 1278.1-1995 standard, Euler Angles are defined as the successive rotations needed to transform from the world (space-fixed) coordinate system to the entity (body-fixed) coordinate system. In that standard, the world coordinate system is the WGS84 Geocentric SRF and the entity coordinate system is as shown in Figure 7, with coordinate system origin at the center of the entity bounding volume.

The specified rotation sequence is the Tait-Bryan rotation with respect to the body-fixed (rotating) frame, shown in Equation (1.15), or the space-fixed (static) equivalent shown in Equation (1.16). To express a world frame coordinate in the entity coordinate frame, the transpose of that rotation is required. The transpose is

$$\begin{aligned} &\boldsymbol{\Omega}_x(\varphi)\,\boldsymbol{\Omega}_y(\theta)\,\boldsymbol{\Omega}_z(\psi) \quad \text{space-fixed} \\ &\boldsymbol{\Omega}_{\tilde{z}}(\psi)\,\boldsymbol{\Omega}_{\tilde{y}}(\theta)\,\boldsymbol{\Omega}_{\tilde{x}}(\phi) \quad \text{body-fixed} \end{aligned} \tag{1.17}$$

The corresponding matrix operator is denoted in the IEEE 1278.1-1995 standard as $[\boldsymbol{R}]_{w\to b}$.

3.2.4.3 Gimbal lock

The term gimbal lock refers to a gyroscope mounted in tree nested gimbals to provide three degree of rotation freedom. Each mounting scheme corresponds to an Euler angle sequence. In any such mounting scheme, there exists critical angles for the middle gimbal that reduce the rotational degrees of freedom to two. The loss of a degree of freedom is termed "gimbal lock".

The case of Euler angles in the z-x-z convention is illustrated by a spinning table top. The top spins on its spin axis and precesses about the precession axis. The angle between the spin and precession axes in the nutation angle. When the spin axis is perfectly vertical (either upright or upside down), the nutation angle is 0 or π and the spin and precession axes become indistinguishable fro each other. This is represented mathematically in Table 4 and Table 5. In those tables, the cases of $\beta = \pi$ or $\beta = 0$ are singular in that only the sum (or difference) of the other angles can be determined.

The case of Euler angles in the x-y-z convention (Tait-Bryan) is illustrated by an aircraft as in Figure 7. When the aircraft either climbs vertically, or dives vertically, the roll axis

---

[3] In this axis assignment, positive pitch tilts the aircraft up (angle of attack), and if the x-axis aligns with local North, yaw corresponds to heading and azimuth.

cannot be distinguished from the yaw axis. This occurs at critical roll angles of $\pm\pi/2$. This is represented mathematically in Table 1 and Table 2. In those tables, the cases of $\beta = \pm\pi/2$ are singular in that only the sum (or difference) of the other angles can be determined.

3.2.5 Quaternions

In this section the definition of quaternions is presented. It is then shown that each quaternion induces a rotation operator. The importance of the algebraic structure of the quaternions is that rotations behave well under these operations.

3.2.5.1 Quaternion notations and conventions

The quaternions are a 4-dimensional vector space together with a vector multiplication operation that forms a non-commutative associative algebra. In analogy to complex numbers that are written as $a + \boldsymbol{i}b$, $\boldsymbol{i}^2 = -1$, quaternion axes $\boldsymbol{i}, \boldsymbol{j}, \boldsymbol{k}$, are defined with the following relationships: $\boldsymbol{i}^2 = \boldsymbol{j}^2 = \boldsymbol{k}^2 = \boldsymbol{ijk} = -1$. A quaternion $\boldsymbol{q}$ is denoted as $\boldsymbol{q} = e_0 + e_1\boldsymbol{i} + e_2\boldsymbol{j} + e_3\boldsymbol{k}$. The first term e_0 is called the “real” (or “scalar”) part of $\boldsymbol{q}$ and $e_1\boldsymbol{i} + e_2\boldsymbol{j} + e_3\boldsymbol{k}$ is called the “imaginary” (or “vector”) part of $\boldsymbol{q}$.

There are several other conventions used to denote a quaternion. To distinguish conventions in this document, the $e_0 + e_1\boldsymbol{i} + e_2\boldsymbol{j} + e_3\boldsymbol{k}$ form will be called the *Hamilton form*. The *scalar vector form* uses an ordered pair of a scalar and 3-tuple vector $\boldsymbol{q} = (e_0, \boldsymbol{e})$. The scalar is the real part of $\boldsymbol{q}$ and the vector corresponds to the imaginary part of $\boldsymbol{q}$, $\boldsymbol{e} = (e_1, e_2, e_3)^{\mathrm{T}}$. As can be seen below, this form allows for some compact notation. (NOTE: In the literature, the order is sometimes reversed: $\boldsymbol{q} = (\boldsymbol{e}, e_0)$.)

Another convention the *4-tuple form* which is just the 4-tuple of numbers $\boldsymbol{q} = (e_0, e_1, e_2, e_3)$. Formulations below will be given in each of these three notational conventions. (NOTE: In the literature, the real part is sometimes placed last: $\boldsymbol{q} = (e_1, e_2, e_3, e_4)$ where $e_4 = e_0$.)

3.2.5.2 Quaternion algebra

Let $p = d_0 + d_1\boldsymbol{i} + d_2\boldsymbol{j} + d_3\boldsymbol{k}$ and $\boldsymbol{q}$ be two quaternions and let t be a scalar. Quaternion addition and scalar multiplication (in each notational convention) is defined as usual for 4D vector space:

$$\begin{aligned}
\boldsymbol{p} + t\boldsymbol{q} &= (d_0 + te_0) + (d_1 + te_1)\,\boldsymbol{i} + (d_2 + te_2)\,\boldsymbol{j} + (d_3 + te_3)\,\boldsymbol{k} && \text{[Hamiltion form]} \\
&= (d_0 + te_0,\ \boldsymbol{d} + t\boldsymbol{e}) && \text{[scalar vector form]} \\
&= (d_0 + te_0,\ d_1 + te_1,\ d_2 + te_2,\ d_3 + te_3) && \text{[4-tuple form]}
\end{aligned}$$

Assuming associative multiplication, the quaternion axes relationships gives the quaternion multiplication rule (in each notational convention):

$$\begin{aligned} \boldsymbol{pq} &= \left(d_0e_0 - d_1e_1 - d_2e_2 - d_3e_3\right) \\ &\quad + \left(d_1e_0 + d_0e_1 + d_2e_3 - d_3e_2\right)\boldsymbol{i} \\ &\quad + \left(d_2e_0 + d_0e_2 + d_3e_1 - d_1e_3\right)\boldsymbol{j} \\ &\quad + \left(d_3e_0 + d_0e_3 + d_1e_2 - d_2e_1\right)\boldsymbol{k} \end{aligned}$$

[Hamiltion form]

$$= \left(\left(d_0e_0 - \boldsymbol{d} \bullet \boldsymbol{e}\right), \left(e_0\boldsymbol{d} + d_0\boldsymbol{e} + \boldsymbol{d} \times \boldsymbol{e}\right)\right)$$

[Scalar vector form]

$$= \begin{pmatrix} \left(d_0e_0 - d_1e_1 - d_2e_2 - d_3e_3\right), \\ \left(d_1e_0 + d_0e_1 + d_2e_3 - d_3e_2\right), \\ \left(d_2e_0 + d_0e_2 + d_3e_1 - d_1e_3\right), \\ \left(d_3e_0 + d_0e_3 + d_1e_2 - d_2e_1\right) \end{pmatrix}$$

[4-tuple form]

(1.18)

Quaternion multiplication is not commutative (note the cross product term in the second notation form). However, the quaternion addition and multiplication operations form an associative algebra.

The conjugate of a quaternion $\boldsymbol{q}$ is defined analogously with complex numbers:

$$\begin{aligned} \boldsymbol{q}^* &= e_0 - e_1\boldsymbol{i} - e_2\boldsymbol{j} - e_3\boldsymbol{k} && \text{[Hamiltion form]} \\ &= \left(e_0, -\boldsymbol{e}\right) && \text{[scalar vector form]} \\ &= \left(e_0, -e_1, -e_2, -e_3\right) && \text{[4-tuple form]} \end{aligned}$$

(1.19)

The product of a quaternion with its conjugate is real and is called the *norm* of $\boldsymbol{q}$:

$$\begin{aligned} \boldsymbol{qq}^* = \boldsymbol{q}^*\boldsymbol{q} &= e_0^2 + e_1^2 + e_2^2 + e_3^2 && \text{[Hamiltion form]} \\ &= \left(e_0^2 + e_1^2 + e_2^2 + e_3^2,\ \mathbf{0}\right) && \text{[scalar vector form]} \\ &= \left(e_0^2 + e_1^2 + e_2^2 + e_3^2,\ 0,\ 0,\ 0\right) && \text{[4-tuple form]} \end{aligned}$$

The *modulus* of a quaternion is defined as the square root of the norm:

$$\left|\boldsymbol{q}\right| = \sqrt{\boldsymbol{qq}^*} = \sqrt{e_0^2 + e_1^2 + e_2^2 + e_3^2}\ .$$

A quaternion $\boldsymbol{q}$ is a *unit quaternion* if $|\boldsymbol{q}| = 1$. In that case $\boldsymbol{q}\boldsymbol{q}^* = \boldsymbol{q}^*\boldsymbol{q} = 1$ which implies that, for a unit quaternion, its conjugate is its multiplicative inverse $\boldsymbol{q}^{-1} = \boldsymbol{q}^*$. More generally, the inverse of a quaternion $\boldsymbol{p}$ is $\boldsymbol{p}^{-1} = \dfrac{\boldsymbol{p}^*}{\boldsymbol{p}\boldsymbol{p}^*} = \dfrac{\boldsymbol{p}^*}{|\boldsymbol{p}|^2}$.

If $\boldsymbol{q} = (e_0, \boldsymbol{e})$ is a unit quaternion, then $\boldsymbol{q}$ may be expressed in the form:

$$\boldsymbol{q} = (\cos(\alpha), \sin(\alpha)\boldsymbol{n})$$

where:

$$\begin{aligned}
&\boldsymbol{n} = \frac{1}{\|\boldsymbol{e}\|}\boldsymbol{e} \text{ is a unit vector in 3D space,} \\
&\alpha = \operatorname{arctan2}(\|\boldsymbol{e}\|, e_0) \text{, and} \\
&\|\boldsymbol{e}\| = \sqrt{e_1^2 + e_2^2 + e_3^2}
\end{aligned} \tag{1.20}$$

Note: Arctan2 is defined in [SRM] sub-clause A.8.1.

3.2.5.3 Quaternion operators on 3D Euclidean space

Each quaternion $\boldsymbol{q}$ corresponds to a linear operator on 3D Euclidean space as follows:

Let $\boldsymbol{r} = (r_1, r_2, r_3)$ be a point in 3D Euclidean space, then the corresponding quaternion is formed by using 0 for the real part and $\boldsymbol{r}$ for the complex part $(0, \boldsymbol{r})$. A unit quaternion $\boldsymbol{q}$ operates on $(0, \boldsymbol{r})$ by left multiplying with $\boldsymbol{q}$ and right multiplying with its conjugate $\boldsymbol{q}^*$. It is shown in Appendix C that the real part of the product $\boldsymbol{q}(0, \boldsymbol{r})q^* = (r_0', \boldsymbol{r}')$, is 0. Thus, $\boldsymbol{q}(0, \boldsymbol{r})q^* = (0, \boldsymbol{r}')$ and the quaternion $\boldsymbol{q}$ associates $\boldsymbol{r}'$ with $\boldsymbol{r}$. Symbolically the operation on $\boldsymbol{r}$ is: $\boldsymbol{r} \mapsto \boldsymbol{r}' = imaginary\ \ part\{\boldsymbol{q}(0, \boldsymbol{r})\boldsymbol{q}^*\}$.

In terms of $\boldsymbol{q} = (e_0, \boldsymbol{e})$ it is also shown in Appendix C that

$$\boldsymbol{r}' = (e_0^2 - \boldsymbol{e} \bullet \boldsymbol{e})\boldsymbol{r} + 2(\boldsymbol{e} \bullet \boldsymbol{r})\boldsymbol{e} + 2e_0\boldsymbol{e} \times \boldsymbol{r}.$$

Since $\boldsymbol{q}$ is a unit quaternion, there exists (equation (1.20)) an angle α satisfying $\boldsymbol{q} = (\cos(\alpha), \sin(\alpha)\boldsymbol{n})$, then by substitution:

$$\boldsymbol{r}' = \left(\cos^2(\alpha) - \sin^2(\alpha)\boldsymbol{n} \bullet \boldsymbol{n}\right)\boldsymbol{r} + 2\sin^2(\alpha)(\boldsymbol{n} \bullet \boldsymbol{r})\boldsymbol{n} + 2\cos(\alpha)\sin(\alpha)\boldsymbol{n} \times \boldsymbol{r}$$

if $\theta = 2\alpha$, then

$\cos(\theta) = \cos^2(\alpha) - \sin^2(\alpha) = 1 - 2\sin^2(\alpha)$, and

$\sin(\theta) = 2\cos(\alpha)\sin(\alpha)$, so that

$$\boldsymbol{r}' = \cos(\theta)\boldsymbol{r} + \left(1 - \cos(\theta)\right)(\boldsymbol{n} \bullet \boldsymbol{r})\boldsymbol{n} + \sin(\theta)\boldsymbol{n} \times \boldsymbol{r}$$

This last expression is Rodrigues' rotation formula (equation (1.3)) for a counter-clockwise rotation about axis $\boldsymbol{n}$ through angle θ, thus quaternion operation on $\boldsymbol{r}$ is a rotation operation. Unit quaternions in scalar vector form are often written as $\boldsymbol{q} = \left(\cos\left(\frac{\theta}{2}\right), \sin\left(\frac{\theta}{2}\right)\boldsymbol{n}\right)$ to indicate the corresponding rotation angle θ (see equation (1.20)).

Let $\boldsymbol{p}$ be any non-zero quaternion and let $\boldsymbol{q} = \frac{\boldsymbol{p}}{|\boldsymbol{p}|^2}$ be its corresponding unit quaternion, then $\boldsymbol{p}(0,\boldsymbol{r})\boldsymbol{p}^{-1} = \boldsymbol{p}(0,\boldsymbol{r})\frac{\boldsymbol{p}^*}{|\boldsymbol{p}|^2} = \frac{\boldsymbol{p}}{|\boldsymbol{p}|}(0,\boldsymbol{r})\frac{\boldsymbol{p}^*}{|\boldsymbol{p}|} = \boldsymbol{q}(0,\boldsymbol{r})\boldsymbol{q}^*$.

This shows that any non-zero quaternion performs a rotation with the $\boldsymbol{p}(0,\boldsymbol{r})\boldsymbol{p}^{-1}$ operation and that this rotation is identical to the rotation performed by the corresponding unit quaternion $\boldsymbol{q}(0,\boldsymbol{r})\boldsymbol{q}^{-1} = \boldsymbol{q}(0,\boldsymbol{r})\boldsymbol{q}^*$. For this reason, some authors use $\boldsymbol{p}(0,\boldsymbol{r})\boldsymbol{p}^{-1}$ operations for any non-zero quaternion while other restrict the set to unit quaternions only and use the $\boldsymbol{q}(0,\boldsymbol{r})\boldsymbol{q}^*$ operator. A unit quaternion in 4-tuple form is also called the *Euler parameters* of a rotation.

The quaternion representation of rotation facilitates the computation the composition of two rotations. If $\boldsymbol{q}_1$ and $\boldsymbol{q}_2$ are two unit quaternions, the composite rotation on $\boldsymbol{r}$ is obtained by first rotating with the rotation operation induced by $\boldsymbol{q}_1$ and then rotating the result with the rotation operation induced by $\boldsymbol{q}_2$ is the same as the single rotation induced by the quaternion product $\boldsymbol{q}_2\boldsymbol{q}_1$ since

$$\boldsymbol{q}_2\left\{\boldsymbol{q}_1(0,\boldsymbol{r})\boldsymbol{q}_1^*\right\}\boldsymbol{q}_2^* = \boldsymbol{q}_2\boldsymbol{q}_1(0,\boldsymbol{r})\boldsymbol{q}_1^*\boldsymbol{q}_2^* = \left\{\boldsymbol{q}_2\boldsymbol{q}_1\right\}(0,\boldsymbol{r})\left\{\boldsymbol{q}_2\boldsymbol{q}_1\right\}^*.$$

3.2.5.4 Quaternions in matrix form

A quaternion $\boldsymbol{q} = (e_0, e_1, e_2, e_3)$ can also be represented as a 2x2 complex matrix or a 4x4 real matrix.

The complex from is $\begin{pmatrix} e_0 + ie_1 & e_2 + ie_3 \\ -e_2 + ie_3 & e_0 - ie_1 \end{pmatrix}$.

The real form is $\begin{pmatrix} e_0 & -e_1 & e_3 & -e_2 \\ e_1 & e_0 & -e_2 & -e_3 \\ -e_3 & e_2 & e_0 & -e_1 \\ e_2 & e_3 & e_1 & e_0 \end{pmatrix}$.

The advantage of these forms are that quaternion addition and multiplication is just matrix addition and matrix multiplication. The conjugate $\boldsymbol{q}^*$ is just the matrix conjugate transpose in the complex case and the matrix transpose in the real case.

3.3 Performing a rotation on an arbitrary vector (formulae)

3.3.1 Rotation about the origin

Rotating a point $\boldsymbol{r}$: Denote the rotated point by $\boldsymbol{r}'$.

Using the rotation matrix form:

The rotated point is obtained by matrix multiplication.

$$\boldsymbol{r}' = \boldsymbol{R}\,\boldsymbol{r}$$

For Euler angles in the z-x-z convention, use transpose of the matrix in Equation (1.10).

For Tait-Bryan angles, use the matrix in Equation (1.8).

Axis-angle form:

A counter-clockwise rotation about axis $\boldsymbol{n}$ (a unit vector) through angle θ is given by Rodrigues' rotation formula:

$$\boldsymbol{r}' = \cos(\theta)\boldsymbol{r} + (1-\cos(\theta))(\boldsymbol{n} \bullet \boldsymbol{r})\boldsymbol{n} + \sin(\theta)\boldsymbol{n} \times \boldsymbol{r} .$$

Using the quaternion form:

$$\boldsymbol{r}' = \left(e_0^2 - e_1^2 - e_2^2 - e_3^2\right)\boldsymbol{r} + 2(\boldsymbol{e} \bullet \boldsymbol{r})\boldsymbol{e} + 2e_0\boldsymbol{r} \times \boldsymbol{e}$$

3.3.2 Rotation about another point

To perform a rotation of point $\boldsymbol{q}$ about the point $\boldsymbol{p}$ to obtain a rotated point $\boldsymbol{q}'$, let

$$\boldsymbol{r} = \boldsymbol{q} - \boldsymbol{p}.$$

Rotate $\boldsymbol{r}$ to $\boldsymbol{r}'$, and then let

$$\boldsymbol{q}' = \boldsymbol{r}' + \boldsymbol{p}.$$

3.4 Inter-converting between representations (formulae)

3.4.1 Euler angles to matrix form

The orientation matrix $\boldsymbol{\Omega}$ is the matrix product of the corresponding three principal orientation matrices defined in 3.2.2, equation (1.7).

For Euler angles in the z-x-z convention:

$$\boldsymbol{\Omega} = \boldsymbol{\Omega}_z(\gamma)\,\boldsymbol{\Omega}_x(\beta)\,\boldsymbol{\Omega}_z(\alpha).$$

The resulting orientation matrix is given in Equation (1.10).

The rotation matrix $\boldsymbol{R}$ is the matrix product of the corresponding three principal rotation matrices defined in 3.2.2, equation (1.7).

$$\boldsymbol{R} = \boldsymbol{R}_z(\alpha)\,\boldsymbol{R}_x(\beta)\,\boldsymbol{R}_z(\gamma).$$

For Euler angles in the x–y–z convention (Tait-Bryan and IEEE 1278.1-1995 Convention) the orientation matrix is:

$$\boldsymbol{\Omega} = \boldsymbol{\Omega}_x(\varphi)\,\boldsymbol{\Omega}_y(\theta)\,\boldsymbol{\Omega}_z(\psi).$$

Substituting angle symbols, $\alpha \leftrightarrow \varphi$, $\beta \leftrightarrow \theta$, $\gamma \leftrightarrow \psi$ in Equation (1.9) gives the resulting orientation matrix.

The rotation matrix is:

$$\boldsymbol{R}_z(\gamma)\,\boldsymbol{R}_y(\beta)\,\boldsymbol{R}_x(\alpha).$$

Substituting angle symbols, $\alpha \leftrightarrow \varphi$, $\beta \leftrightarrow \theta$, $\gamma \leftrightarrow \psi$ in Equation (1.8) gives the resulting rotation matrix.

3.4.2 Matrix to axis-angle

Given a rotation matrix $\boldsymbol{R} = \begin{pmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{pmatrix}$, find unit vector $\boldsymbol{n}$ and angle θ so that $(\boldsymbol{n}, \theta)$ is the axis-angle representation of the rotation.

The formulations here are derived in Appendix E where it is shown that:

$$\theta = \arccos\left(\left(\frac{1-\mathrm{Trace}(\boldsymbol{R})}{2}\right)\right) = \arccos\left(\left(\frac{1-(r_{11}+r_{22}+r_{33})}{2}\right)\right),\ \ 0 \le \theta \le \pi.$$

There are three cases for the computation of $\boldsymbol{n}$ that depend on the value of θ.

Case $\theta = 0$: There is no rotation so $\boldsymbol{n}$ is in-determinant.

Case $0 < \theta < \pi$: Let $\boldsymbol{n} = \frac{1}{\|\boldsymbol{v}\|}\boldsymbol{v}$, where:

$$\boldsymbol{v} = \begin{pmatrix} r_{32} - r_{23} \\ r_{13} - r_{31} \\ r_{21} - r_{12} \end{pmatrix}.$$

Case: $\theta = \pi$: First find the maximum diagonal element a_{11}, a_{22}, or a_{33} of $\boldsymbol{R}$. Then:

Sub-case a_{11} is the maximum: Let $v_1 = \sqrt{(a_{11}+1)/2},\ v_2 = \frac{a_{12}}{2v_1},\ v_3 = \frac{a_{13}}{2v_1}$.

Sub-case a_{22} is the maximum: Let $v_2 = \sqrt{(a_{22}+1)/2},\ v_1 = \frac{a_{12}}{2v_2},\ v_3 = \frac{a_{32}}{2v_2}$.

Sub-case a_{33} is the maximum: Let $v_3 = \sqrt{(a_{33}+1)/2},\ v_1 = \frac{a_{13}}{2v_3},\ v_2 = \frac{a_{23}}{2v_3}$

Finally let $\boldsymbol{n} = \frac{1}{\|\boldsymbol{v}\|}\boldsymbol{v}$, where $\boldsymbol{v} = (v_1, v_2, v_3)^{\mathrm{T}}$.

Given an orientation matrix $\boldsymbol{\Omega}$, let $\boldsymbol{R} = \boldsymbol{\Omega}^{\mathrm{T}}$ and compute $(\boldsymbol{n}, \theta)$ as above.

3.4.3 Axis-angle to rotation matrix

To convert rotation axis $\boldsymbol{n}$ and angle θ form $(\boldsymbol{n},\theta)$ to the corresponding rotation matrix $\boldsymbol{R}$, use Rodrigues' rotation formula (equation (1.4) or (1.5)):

$$\begin{aligned} \boldsymbol{R} &= \left[\boldsymbol{I}_{3\times3} + \sin(\theta)\boldsymbol{S}_{\boldsymbol{n}} + (1-\cos(\theta))\boldsymbol{S}_{\boldsymbol{n}}^2\right] \\ &= \left[\cos(\theta)\boldsymbol{I}_{3\times3} + (1-\cos(\theta))\boldsymbol{n}\otimes\boldsymbol{n} + \sin(\theta)\boldsymbol{S}_{\boldsymbol{n}}\right] \end{aligned} \qquad \text{[revised] (1.21)}$$

where:

$$\boldsymbol{S}_{\boldsymbol{n}} = \begin{pmatrix} 0 & -n_3 & n_2 \\ n_3 & 0 & -n_1 \\ -n_2 & n_1 & 0 \end{pmatrix}$$

is the skew-symmetric matrix associated with $\boldsymbol{n}$.

The form of $\boldsymbol{R}$ expanded to matrix elements is:

$$\begin{pmatrix} (1-\cos\theta)n_1^2+\cos\theta & (1-\cos\theta)n_1n_2-n_3\sin\theta & (1-\cos\theta)n_1n_3+n_2\sin\theta \\ (1-\cos\theta)n_2n_1+n_3\sin\theta & (1-\cos\theta)n_2^2+\cos\theta & (1-\cos\theta)n_2n_3-n_1\sin\theta \\ (1-\cos\theta)n_3n_1-n_2\sin\theta & (1-\cos\theta)n_3n_2+n_1\sin\theta & (1-\cos\theta)n_3^2+\cos\theta \end{pmatrix} \tag{1.22}$$

3.4.4 Axis-angle to quaternion

Starting with axis-angle form $(\boldsymbol{n},\theta)$, where $\boldsymbol{n}$ is a unit vector and angle θ is a counter-clockwise rotation about $\boldsymbol{n}$. Let:

$$\begin{aligned} e_0 &= \cos\left(\frac{\theta}{2}\right) \\ \boldsymbol{e} &= \begin{pmatrix} e_1 \\ e_2 \\ e_3 \end{pmatrix} = \sin\left(\frac{\theta}{2}\right)\boldsymbol{n} = \sin\left(\frac{\theta}{2}\right)\begin{pmatrix} n_1 \\ n_2 \\ n_3 \end{pmatrix} \end{aligned} \tag{1.23}$$

Then the corresponding quaternion is:

$$\begin{aligned} \boldsymbol{q} &= e_0 + e_1\boldsymbol{i} + e_2\boldsymbol{j} + e_3\boldsymbol{k} && \text{[Hamiltion form]} \\ &= (e_0, \boldsymbol{e}) && \text{[scalar vector form]} \\ &= (e_0, e_1, e_2, e_3) && \text{[4-tuple form]} \end{aligned} \tag{1.24}$$

3.4.5 Rotation matrix to quaternion

Convert the matrix to axis-angle form as in 3.4.2. Then convert the axis-angle form to quaternion form as in 3.4.4.

3.4.6 Quaternion to rotation matrix

Given:

$$\begin{aligned} \boldsymbol{q} &= e_0 + e_1\boldsymbol{i} + e_2\boldsymbol{j} + e_3\boldsymbol{k} && \text{[Hamiltion form]} \\ &= (e_0, \boldsymbol{e}) && \text{[scalar vector form]} \\ &= (e_0, e_1, e_2, e_3) && \text{[4-tuple form]} \end{aligned}$$

the rotation matrix is then:

$$\boldsymbol{R} = \begin{pmatrix} 1-2\left(e_2^2+e_3^2\right) & 2\left(e_1e_2-e_0e_3\right) & 2\left(e_1e_3+e_0e_2\right) \\ 2\left(e_1e_2+e_0e_3\right) & 1-2\left(e_1^2+e_3^2\right) & 2\left(e_2e_3-e_0e_1\right) \\ 2\left(e_1e_3-e_0e_2\right) & 2\left(e_2e_3+e_0e_1\right) & 1-2\left(e_1^2+e_2^2\right) \end{pmatrix} \tag{1.25}$$

Since $e_0^2+e_1^2+e_2^2+e_3^2=1$, the diagonal terms in the matrix can be re-written in the following equivalent form:

$$\boldsymbol{R} = \begin{pmatrix} e_0^2+e_1^2-e_2^2-e_3^2 & 2\left(e_1e_2-e_0e_3\right) & 2\left(e_1e_3+e_0e_2\right) \\ 2\left(e_1e_2+e_0e_3\right) & e_0^2-e_1^2+e_2^2-e_3^2 & 2\left(e_2e_3-e_0e_1\right) \\ 2\left(e_1e_3-e_0e_2\right) & 2\left(e_2e_3+e_0e_1\right) & e_0^2-e_1^2-e_2^2+e_3^2 \end{pmatrix} \tag{1.26}$$

It is also shown in Appendix C that

$$\boldsymbol{r}' = \left(e_0^2-\boldsymbol{e}\bullet\boldsymbol{e}\right)\boldsymbol{r}+2\left(\boldsymbol{e}\bullet\boldsymbol{r}\right)\boldsymbol{e}+2e_0\boldsymbol{e}\times\boldsymbol{r}.$$

This equation in matrix form is:

$$\boldsymbol{r}' = \left[\left(e_0^2-e_1^2-e_2^2-e_3^2\right)\boldsymbol{I}_{3\times3}+2\boldsymbol{e}\otimes\boldsymbol{e}+2e_0\boldsymbol{S_e}\right]\boldsymbol{r}$$

where $\boldsymbol{S_e} = \begin{pmatrix} 0 & -e_3 & e_2 \\ e_3 & 0 & -e_1 \\ -e_2 & e_1 & 0 \end{pmatrix}$

The expansion of this expression gives equation (1.26).

3.4.7 Quaternion to Axis-angle

Given a unit quaternion $\boldsymbol{q}=\left(e_0,\boldsymbol{e}\right)$, find the corresponding axis angle form $\left(\boldsymbol{n},\theta\right)$.

Section 3.4.4 shows that the quaternion corresponding to axis angle $\left(\boldsymbol{n},\theta\right)$ is

$$\boldsymbol{q} = \left(\cos\left(\frac{\theta}{2}\right),\sin\left(\frac{\theta}{2}\right)\boldsymbol{n}\right).$$

Reversing this formulation yields:

$$(\boldsymbol{n}, \theta) = \left(\boldsymbol{e}/\|\boldsymbol{e}\|,\ 2 * \text{arctan2}\left(\|\boldsymbol{e}\|, e_0\right)\right)$$
$$= \left(\frac{\boldsymbol{e}}{\sqrt{1-e_0^2}},\ 2 * \text{arctan2}\left(\sqrt{1-e_0^2}, e_0\right)\right)$$

3.4.8 Matrix to Euler angles

Given a rotation or orientation matrix $\boldsymbol{M} = \begin{pmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{pmatrix}$, determine the corresponding Euler angles.

To find the Euler angles in the $\boldsymbol{x}$–$\boldsymbol{y}$–$\boldsymbol{z}$ rotation convention $\boldsymbol{M} = \boldsymbol{R_z}(\gamma)\boldsymbol{R_y}(\beta)\boldsymbol{R_x}(\alpha)$, use Table 1.

To find the Euler angles in the $\boldsymbol{x}$–$\boldsymbol{y}$–$\boldsymbol{z}$ orientation convention $\boldsymbol{M} = \boldsymbol{\Omega_z}(\gamma)\boldsymbol{\Omega_y}(\beta)\boldsymbol{\Omega_x}(\alpha)$, use Table 2. In case of Tait-Bryan angles, the following symbol correspondence is made:

$$\alpha \leftrightarrow \phi$$
$$\beta \leftrightarrow \theta$$
$$\gamma \leftrightarrow \psi$$

To find the Euler angles in the $\boldsymbol{z}$–$\boldsymbol{x}$–$\boldsymbol{z}$ orientation convention $\boldsymbol{M} = \boldsymbol{\Omega_z}(\gamma)\boldsymbol{\Omega_x}(\beta)\boldsymbol{\Omega_z}(\alpha)$ use Table 4.

3.4.9 Euler angles to quaternion

The principal rotations (section 3.2.2) correspond to the following quaternions:

$$\boldsymbol{R_z}(\gamma) \leftrightarrow \left(\cos\left(\frac{\gamma}{2}\right), \sin\left(\frac{\gamma}{2}\right)\boldsymbol{z}\right)$$
$$\boldsymbol{R_y}(\beta) \leftrightarrow \left(\cos\left(\frac{\beta}{2}\right), \sin\left(\frac{\beta}{2}\right)\boldsymbol{y}\right)$$
$$\boldsymbol{R_x}(\alpha) \leftrightarrow \left(\cos\left(\frac{\alpha}{2}\right), \sin\left(\frac{\alpha}{2}\right)\boldsymbol{x}\right)$$

For each Euler convention, multiply the corresponding quaternions. Terms in the resulting product may be simplified using the orthonormal property of the vector set $\boldsymbol{x}$, $\boldsymbol{y}$ and $\boldsymbol{z}$.

For Euler angles in the z-x-z convention, the rotation $\boldsymbol{R_z}(\gamma)\boldsymbol{R_x}(\beta)\boldsymbol{R_z}(\alpha)$ corresponds to:

$$\boldsymbol{q} = \left(\cos\left(\frac{\gamma}{2}\right), \sin\left(\frac{\gamma}{2}\right)\boldsymbol{z}\right)\left(\cos\left(\frac{\beta}{2}\right), \sin\left(\frac{\beta}{2}\right)\boldsymbol{x}\right)\left(\cos\left(\frac{\alpha}{2}\right), \sin\left(\frac{\alpha}{2}\right)\boldsymbol{z}\right)$$

When multiplied out with the quaternion multiplication rule (Equation (1.18)), the expression reduces to:

$$\boldsymbol{q} = \left(e_0, \left(e_1, e_2, e_3\right)^{\mathsf{T}}\right)$$

where:

$$\begin{aligned}
e_0 &= \cos\left(\frac{\gamma}{2}\right)\cos\left(\frac{\beta}{2}\right)\cos\left(\frac{\alpha}{2}\right) - \sin\left(\frac{\gamma}{2}\right)\cos\left(\frac{\beta}{2}\right)\sin\left(\frac{\alpha}{2}\right) \\
e_1 &= \cos\left(\frac{\gamma}{2}\right)\sin\left(\frac{\beta}{2}\right)\cos\left(\frac{\alpha}{2}\right) + \sin\left(\frac{\gamma}{2}\right)\sin\left(\frac{\beta}{2}\right)\sin\left(\frac{\alpha}{2}\right) \\
e_2 &= \sin\left(\frac{\gamma}{2}\right)\sin\left(\frac{\beta}{2}\right)\cos\left(\frac{\alpha}{2}\right) - \cos\left(\frac{\gamma}{2}\right)\sin\left(\frac{\beta}{2}\right)\sin\left(\frac{\alpha}{2}\right) \\
e_3 &= \sin\left(\frac{\gamma}{2}\right)\cos\left(\frac{\beta}{2}\right)\cos\left(\frac{\alpha}{2}\right) - \cos\left(\frac{\gamma}{2}\right)\cos\left(\frac{\beta}{2}\right)\sin\left(\frac{\alpha}{2}\right)
\end{aligned}$$

For Euler angles the in the Tait-Bryan x–y–z convention, the rotation $\boldsymbol{R_x}(\phi)\boldsymbol{R_y}(\theta)\boldsymbol{R_z}(\psi)$ corresponds to:

$$\boldsymbol{q} = \left(\cos\left(\frac{\phi}{2}\right), \sin\left(\frac{\phi}{2}\right)\boldsymbol{x}\right)\left(\cos\left(\frac{\theta}{2}\right), \sin\left(\frac{\theta}{2}\right)\boldsymbol{y}\right)\left(\cos\left(\frac{\psi}{2}\right), \sin\left(\frac{\psi}{2}\right)\boldsymbol{z}\right)$$

When multiplied out, the expression reduces to:

$$\boldsymbol{q} = \left(e_0, \left(e_1, e_2, e_3\right)^{\mathsf{T}}\right)$$

where:

$$e_0 = \cos\left(\frac{\phi}{2}\right)\cos\left(\frac{\theta}{2}\right)\cos\left(\frac{\psi}{2}\right) - \sin\left(\frac{\phi}{2}\right)\sin\left(\frac{\theta}{2}\right)\sin\left(\frac{\psi}{2}\right)$$
$$e_1 = \cos\left(\frac{\phi}{2}\right)\sin\left(\frac{\theta}{2}\right)\sin\left(\frac{\psi}{2}\right) + \cos\left(\frac{\psi}{2}\right)\cos\left(\frac{\theta}{2}\right)\sin\left(\frac{\phi}{2}\right)$$
$$e_2 = \cos\left(\frac{\phi}{2}\right)\sin\left(\frac{\theta}{2}\right)\cos\left(\frac{\psi}{2}\right) - \sin\left(\frac{\psi}{2}\right)\cos\left(\frac{\theta}{2}\right)\sin\left(\frac{\phi}{2}\right)$$
$$e_3 = \cos\left(\frac{\phi}{2}\right)\cos\left(\frac{\theta}{2}\right)\sin\left(\frac{\psi}{2}\right) + \cos\left(\frac{\psi}{2}\right)\sin\left(\frac{\theta}{2}\right)\sin\left(\frac{\phi}{2}\right)$$

The IEEE 1278.1-1995 Convention orientation $\boldsymbol{\Omega_x}(\phi)\,\boldsymbol{\Omega_y}(\theta)\,\boldsymbol{\Omega_z}(\psi)$ corresponds to:

$$\boldsymbol{q} = \left(\cos\left(\frac{-\phi}{2}\right), \sin\left(\frac{-\phi}{2}\right)\boldsymbol{x}\right)\left(\cos\left(\frac{-\theta}{2}\right), \sin\left(\frac{-\theta}{2}\right)\boldsymbol{y}\right)\left(\cos\left(\frac{-\psi}{2}\right), \sin\left(\frac{-\psi}{2}\right)\boldsymbol{z}\right)$$

When multiplied out, the expression reduces to:

$$\boldsymbol{q} = \left(e_0, (e_1, e_2, e_3)^{\mathsf{T}}\right)$$

where:

$$e_0 = \cos\left(\frac{\phi}{2}\right)\cos\left(\frac{\theta}{2}\right)\cos\left(\frac{\psi}{2}\right) + \sin\left(\frac{\phi}{2}\right)\sin\left(\frac{\theta}{2}\right)\sin\left(\frac{\psi}{2}\right)$$
$$e_1 = \cos\left(\frac{\phi}{2}\right)\sin\left(\frac{\theta}{2}\right)\sin\left(\frac{\psi}{2}\right) - \cos\left(\frac{\psi}{2}\right)\cos\left(\frac{\theta}{2}\right)\sin\left(\frac{\phi}{2}\right)$$
$$e_2 = -\cos\left(\frac{\phi}{2}\right)\sin\left(\frac{\theta}{2}\right)\cos\left(\frac{\psi}{2}\right) - \sin\left(\frac{\psi}{2}\right)\cos\left(\frac{\theta}{2}\right)\sin\left(\frac{\phi}{2}\right)$$
$$e_3 = -\cos\left(\frac{\phi}{2}\right)\cos\left(\frac{\theta}{2}\right)\sin\left(\frac{\psi}{2}\right) + \cos\left(\frac{\psi}{2}\right)\sin\left(\frac{\theta}{2}\right)\sin\left(\frac{\phi}{2}\right)$$

3.4.10 various to Euler angles

3.5 Considerations for computational and storage efficiency

[Possible future sections that compares various representation with respect to:

1. minimal storage size,
2. storage with cached values (for computational efficiency),

3. counts of floating point instruction steps for various operations.

]

3.6 Interpolation issues

[Possible future section that deals with quaternion interpolation between two orientations.]

3.7 Error analysis

[Possible future section that investigates change rates in various representations]

4 Rotational Kinematics

4.1 Rotational velocity and acceleration

Consider first the special case of rotation of a point about a fixed unit vector axis $\boldsymbol{n}$ as a differentiable function of time, $\boldsymbol{r}(t)=\boldsymbol{R}_{\boldsymbol{n}}(\theta(t))\boldsymbol{r}_0$. Assume that the rotation begins at $t=0$ so that $\theta(0)=0$ and $\boldsymbol{r}(0)=\boldsymbol{r}_0$. By the alternate form of Rodrigues' rotation formula:

$$\boldsymbol{r}(t)=\boldsymbol{R}_{\boldsymbol{n}}(\theta(t))\boldsymbol{r}_0=\boldsymbol{r}_0+(1-\cos(\theta))\boldsymbol{n}\times(\boldsymbol{n}\times\boldsymbol{r}_0)+\sin(\theta)\boldsymbol{n}\times\boldsymbol{r}_0 .$$

The rotational velocity at time t is then:

$$\begin{aligned}\frac{d}{dt}\boldsymbol{r}(t)&=\frac{d}{dt}\left(\boldsymbol{r}_0+(1-\cos(\theta))\boldsymbol{n}\times(\boldsymbol{n}\times\boldsymbol{r}_0)+\sin(\theta)\boldsymbol{n}\times\boldsymbol{r}_0\right)\\&=\sin(\theta)\frac{d\theta}{dt}\boldsymbol{n}\times(\boldsymbol{n}\times\boldsymbol{r}_0)+\cos(\theta)\frac{d\theta}{dt}\boldsymbol{n}\times\boldsymbol{r}_0\end{aligned}$$

Thus the velocity is the sum of two vector components. One component $\boldsymbol{n}\times(\boldsymbol{n}\times\boldsymbol{r}_0)$ points toward the rotation axis and the other $\boldsymbol{n}\times\boldsymbol{r}_0$ is tangent to the arc or rotation.

Evaluating at $t=0$ gives the instantaneous rotational velocity:

$$\frac{d}{dt}\boldsymbol{r}(0)=\frac{d\theta}{dt}\boldsymbol{n}\times\boldsymbol{r}_0=\dot{\theta}\,\boldsymbol{n}\times\boldsymbol{r}_0=\boldsymbol{\omega}\times\boldsymbol{r}_0 \text{, where } \boldsymbol{\omega}=\dot{\theta}\,\boldsymbol{n} .$$

The rotational acceleration is the second time derivative:

$$\begin{aligned}\frac{d^2}{dt^2}\boldsymbol{r}(t)&=\frac{d}{dt}\left(\sin(\theta)\frac{d\theta}{dt}\boldsymbol{n}\times(\boldsymbol{n}\times\boldsymbol{r}_0)+\cos(\theta)\frac{d\theta}{dt}\boldsymbol{n}\times\boldsymbol{r}_0\right)\\&=\left(\cos(\theta)\frac{d\theta}{dt}+\sin(\theta)\frac{d^2\theta}{dt^2}\right)\boldsymbol{n}\times(\boldsymbol{n}\times\boldsymbol{r}_0)+\left(-\sin(\theta)\frac{d\theta}{dt}+\cos(\theta)\frac{d^2\theta}{dt^2}\right)\boldsymbol{n}\times\boldsymbol{r}_0.\end{aligned}$$

Evaluating at $t = 0$ gives the instantaneous rotational acceleration $\boldsymbol{\alpha}$:

$$\boldsymbol{\alpha} = \dot{\boldsymbol{\omega}} = \dot{\theta}\,\boldsymbol{n}\times\left(\boldsymbol{n}\times\boldsymbol{r}_0\right) + \ddot{\theta}\,\boldsymbol{n}\times\boldsymbol{r}_0.$$

The first component points towards the rotation axis as is called the centripetal acceleration.

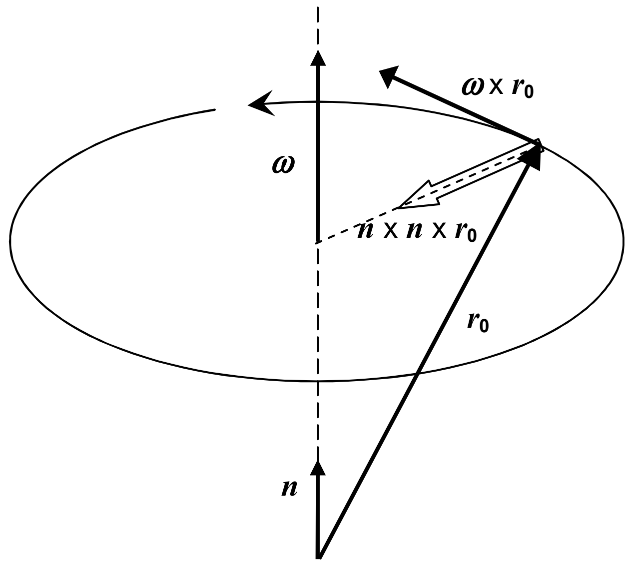

Figure 8

In the general case (in which the rotation axis may vary as function of time), we set $\boldsymbol{r}(t) = \boldsymbol{R}(t)\boldsymbol{r}_0$ with the initial condition $\boldsymbol{R}(0) = \boldsymbol{I}_{3\times3}$ (that is, $\boldsymbol{r}(0) = \boldsymbol{r}_0$). To compute $\dot{\boldsymbol{R}} = \frac{d\boldsymbol{R}(0)}{dt}$, we note that as a rotation matrix we have $\boldsymbol{I}_{3\times3} = \boldsymbol{R}(t)\boldsymbol{R}^{\mathsf{T}}(t)$ so that:

$$\begin{aligned}\frac{d}{dt}\boldsymbol{I}_{3\times3} &= \frac{d}{dt}\left[\boldsymbol{R}(t)\boldsymbol{R}^{\mathsf{T}}(t)\right] \\ \mathbf{0} &= \dot{\boldsymbol{R}}(t)\boldsymbol{R}^{\mathsf{T}}(t) + \boldsymbol{R}(t)\dot{\boldsymbol{R}}^{\mathsf{T}}(t)\end{aligned}$$

At $t = 0$, this expression reduces to $\dot{\boldsymbol{R}}(0) = -\dot{\boldsymbol{R}}^{\mathsf{T}}(0)$, so $\dot{\boldsymbol{R}}(0)$ is skew-symmetric and must be in the form $\dot{\boldsymbol{R}}(0) = \begin{pmatrix} 0 & -\omega_3 & \omega_2 \\ \omega_3 & 0 & -\omega_1 \\ -\omega_2 & \omega_1 & 0 \end{pmatrix} = \boldsymbol{S}_{\boldsymbol{\omega}}$ where $\boldsymbol{\omega} = \left(\omega_1, \omega_2, \omega_3\right)^{\mathsf{T}}$. Thus we have $\frac{d}{dt}\boldsymbol{r}(0) = \boldsymbol{S}_{\boldsymbol{\omega}}\boldsymbol{r} = \boldsymbol{\omega}\times\boldsymbol{r}_0$.

4.2 Orientation (Ω), angular velocity (ω), angular acceleration (α), torque (τ)

We consider three cases ….

Static case: Consider a rigid body with a body-fixed coordinate system origin at some point C on the body and let P be another point on the body with body coordinate vector $\boldsymbol{b}$. Given $\boldsymbol{s}_{\mathrm{C}}$ the space-fixed coordinate vector of C and body orientation $\boldsymbol{\Omega}$, the space coordinate vector $\boldsymbol{s}$ for the point P is computed as $\boldsymbol{s} = \boldsymbol{s}_{\mathrm{C}} + \boldsymbol{\Omega}\,\boldsymbol{b}$. In this expression, the term $\boldsymbol{\Omega}\,\boldsymbol{b}$ is to understood as orientation operator acting on $\boldsymbol{b}$. The implementation of this operation may depend on the specification of the orientation. For example $\boldsymbol{\Omega}\,\boldsymbol{b}$ may be computed:

- by Rodrigues' rotation formula if the orientation was specified in axis-angle form,
- by a quaternion operator if the orientation was specified in quaternion form,
- by matrix multiplication using a matrix computed from an Euler angle specification of the orientation, or
- by matrix multiplication the directional cosine matrix computed from the body coordinate system basis vectors in space coordinates.

This comment applies to all such operations that follow.

Rigid motion case: In this case the body is moving so that $\boldsymbol{s}_{\mathrm{C}}$ and $\boldsymbol{\Omega}$ are functions of time and we have:

$$\boldsymbol{s}(t) = \boldsymbol{s}_{\mathrm{C}}(t) + \boldsymbol{\Omega}(t)\,\boldsymbol{b}$$

$$\dot{\boldsymbol{s}}(t) = \boldsymbol{v}(t) + \boldsymbol{\omega}(t) \times \boldsymbol{b}$$

$$\ddot{\boldsymbol{s}}(t) = \boldsymbol{a}(t) + \boldsymbol{\alpha}(t) \times \boldsymbol{b}$$

where:

$$v(t) = \dot{\boldsymbol{s}}_{\mathrm{C}}(t),$$

$$\boldsymbol{a}(t) = \dot{\boldsymbol{v}}(t),$$

$$\boldsymbol{\omega}(t) = \dot{\boldsymbol{\Omega}}(t), \text{ and}$$

$$\boldsymbol{\alpha}(t) = \dot{\boldsymbol{\omega}}(t).$$

Note that $\boldsymbol{b}$ is time independent because it is fixed on the rigid body. The orientation operator $\boldsymbol{\Omega}$ converts directions from the body coordinate system to the space coordinate system, so that its derivatives $\boldsymbol{\omega}$ and $\boldsymbol{\alpha}$ have space coordinate system values. The corresponding values in body coordinates are obtained with in inverse operator $\boldsymbol{\Omega}^{\mathsf{T}}$:

$$\boldsymbol{\omega}_{\mathrm{B}}(t) = \boldsymbol{\Omega}^{\mathsf{T}}(t)\,\boldsymbol{\omega}(t)$$
$$\boldsymbol{\alpha}_{\mathrm{B}} = \boldsymbol{\Omega}^{\mathsf{T}}(t)\,\boldsymbol{\alpha}(t)$$

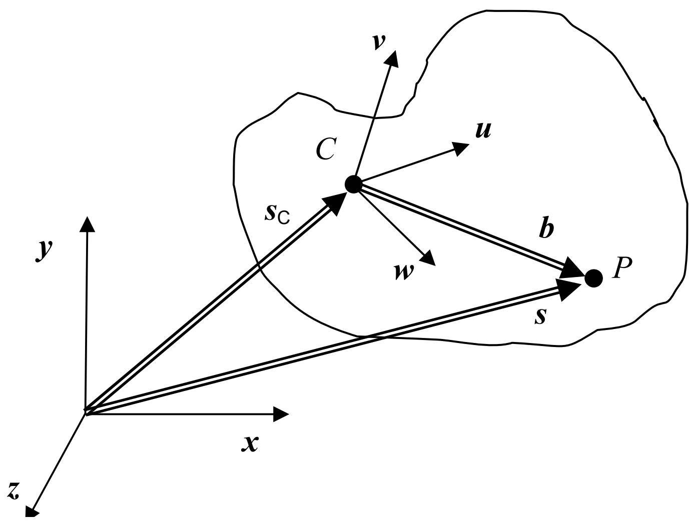

Figure 9

In the more general case, the point P is moving with respect to both coordinate systems.

4.3 Dynamics

4.3.1Rigid body dynamics

Consider a rigid body consisting of discrete particles P_i of mass m_i and/or volume elements $\boldsymbol{V}$ with mass density function[4] ρ. $\boldsymbol{s}_i(t)$ will denote the coordinate vector at time t of point P_i in a space coordinate system with respect to the orthonormal basis $\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{z}$ and $\boldsymbol{b}_i$ will denote the same point in a body coordinate system with respect to the

---

[4] At a point P the mass density $\rho(P)$ may expressed as either function of a space coordinate $\rho_{\mathrm{SPACE}}(\boldsymbol{s})$ or a body coordinate $\rho_{\mathrm{BODY}}(\boldsymbol{b})$. This notational distinction will not employed as the context makes it clear which function is intended.

orthonormal basis $\boldsymbol{u}, \boldsymbol{v}, \boldsymbol{w}$ attached to the body. Because the body is rigid, coordinate vectors $\boldsymbol{b}_i$ are independent of time.

The *total mass* of the body is: $M = \sum_i m_i + \iiint_{\boldsymbol{V}} \rho \; dV$.

The *center of mass* is located at $\boldsymbol{s}_{\mathrm{CM}}(t) = \frac{1}{M}\left(\sum_i m_i \boldsymbol{s}_i(t) + \iiint_{s \in \boldsymbol{V}} \rho(\boldsymbol{s}(t))\, \boldsymbol{s}(t)\; d\boldsymbol{s} \right)$.

We further require that the origin of the body coordinate system coincide with the center of mass. As a consequence, in body coordinates, the center of mass is the body coordinate system zero vector:

$$\mathbf{0} = \frac{1}{M}\left(\sum_i m_i \boldsymbol{b}_i + \iiint_{b \in \boldsymbol{V}} \rho(\boldsymbol{b})\, \boldsymbol{b}\; d\boldsymbol{b} \right)$$

If $\boldsymbol{\Omega}(t)$ is the cosine matrix of $\boldsymbol{u}, \boldsymbol{v}, \boldsymbol{w}$ with respect to $\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{z}$, then

$$\boldsymbol{s}_i(t) = \boldsymbol{s}_{\mathrm{CM}}(t) + \boldsymbol{\Omega}(t)\boldsymbol{b}_i \text{ and } \boldsymbol{b}_i = \boldsymbol{\Omega}^{\mathrm{T}}(t)\left[\boldsymbol{s}_i(t) - \boldsymbol{s}_{\mathrm{CM}}(t) \right],$$

and velocities are given by

$$\begin{aligned}
\boldsymbol{v}_i(t) &= \dot{\boldsymbol{s}}_i(t) \\
&= \boldsymbol{v}(t) + \omega(t) \times \left(\boldsymbol{s}_i(t) - \boldsymbol{s}_{\mathrm{CM}}(t) \right) \\
&= \boldsymbol{v}(t) + \boldsymbol{\Omega}(t)\left(\omega_{\mathrm{B}}(t) \times \boldsymbol{b}_i \right)
\end{aligned}$$

where $\boldsymbol{v}(t) = \dot{\boldsymbol{s}}_{\mathrm{CM}}(t)$ and $\omega(t)$ is the rotational velocity in space coordinates.

The *linear momentum* of a particle is defined by $\boldsymbol{p}_i = m\,\dot{\boldsymbol{s}}_i = m\boldsymbol{v}_i$. The *total linear momentum* of the body is:

$$\begin{aligned}
\boldsymbol{P}(t) &= \sum_i m_i\, \dot{\boldsymbol{s}}_i(t) + \iiint_{s \in \boldsymbol{V}} \rho(\boldsymbol{s}(t))\, \dot{\boldsymbol{s}}(t)\; d\boldsymbol{s} \\
&= \sum_i m_i \left[\dot{\boldsymbol{s}}_{\mathrm{CM}}(t) + \dot{\boldsymbol{\Omega}}(t)\boldsymbol{b}_i \right] + \iiint_{b \in \boldsymbol{V}} \rho(\boldsymbol{b}) \left[\dot{\boldsymbol{s}}_{\mathrm{CM}}(t) + \dot{\boldsymbol{\Omega}}(t)\boldsymbol{b} \right] d\boldsymbol{b} \\
&= \left(\sum_i m_i + \iiint_{b \in \boldsymbol{V}} \rho(\boldsymbol{b})\; d\boldsymbol{b} \right) \dot{\boldsymbol{s}}_{\mathrm{CM}}(t) + \dot{\boldsymbol{\Omega}}(t) \left(\sum_i m_i \boldsymbol{b}_i + \iiint_{b \in \boldsymbol{V}} \rho(\boldsymbol{b})\, \boldsymbol{b}\; d\boldsymbol{b} \right) \\
&= M\, \dot{\boldsymbol{s}}_{\mathrm{CM}}(t) + \dot{\boldsymbol{\Omega}}(t)\mathbf{0} \\
&= M\, \boldsymbol{v}(t)
\end{aligned}$$

This result:

$$\boldsymbol{P} = M\boldsymbol{v}(t)$$

shows that total linear momentum is independent of orientation or rotational velocity.

If a point $\boldsymbol{b}$ with mass m is rotating about an axis with (scalar) rotational rate ω, in a circle of radius r, then its speed is $r\omega$ and its scalar linear momentum is $p = m(r\omega)$. The scalar angular momentum L of $\boldsymbol{b}$ is its scalar linear momentum multiplied by the length r of it moment arm: $L = rp = r^2 m\omega$. The terms in this expression that depend only on the geometric distribution of mass with respect to the rotation axis is the scalar moment of inertia $I = r^2 m$. We then have $L = I\omega$. If the rotational axis is determined by a unit vector $\boldsymbol{n}$ and if the angle between $\boldsymbol{n}$ and $\boldsymbol{b}$ is φ, then we note that the length the moment arm is $r = \|\boldsymbol{b}\| \sin\varphi = \|\boldsymbol{b} \times \boldsymbol{n}\|$. This motivates the following definitions. The rotational momentum of particle P_i is defined as $\boldsymbol{L}_i(t) = \boldsymbol{b}_i \times \boldsymbol{p}_i(t)$. Note that the rotational momentum is a vector quantity that is perpendicular to both the moment arm $\boldsymbol{b}_i$ and the momentum vector $\boldsymbol{p}_i$. The rotational momentum $\boldsymbol{L}_i$ may also be expressed directly in terms of the rotational velocity: $\boldsymbol{L}_i(t) = \boldsymbol{b}_i \times \boldsymbol{p}_i(t) = \boldsymbol{b}_i \times m_i \boldsymbol{v}(t) = m_i \boldsymbol{b}_i \times (\boldsymbol{\omega}_{\mathrm{B}}(t) \times \boldsymbol{b}_i)$.

The total *rotational momentum* of the body is defined as:

$$\boldsymbol{L}_{\mathrm{B}}(t) = \sum_i m_i \boldsymbol{b}_i \times (\boldsymbol{\omega}_{\mathrm{B}}(t) \times \boldsymbol{b}_i) + \iiint \rho(\boldsymbol{b})\, \boldsymbol{b} \times (\boldsymbol{\omega}_{\mathrm{B}}(t) \times \boldsymbol{b})\, d\boldsymbol{b}$$

The subscript B indicates that the vector values of $\boldsymbol{L}_{\mathrm{B}}(t)$ are represented in body space coordinates. Since $\boldsymbol{\omega}_{\mathrm{B}}(t)$ varies in time, all the summation and integration in the above expression needs to be re-evaluated at each time t. However, the above expression is linear in $\boldsymbol{\omega}_{\mathrm{B}}(t)$ and, as shown in Appendix D, it may be factored out to the equivalent form:

$$\boldsymbol{L}_{\mathrm{B}}(t) = \boldsymbol{I}_{\otimes \mathrm{B}} \boldsymbol{\omega}_{\mathrm{B}}(t)$$

where $\boldsymbol{I}_{\otimes \mathrm{B}}$ is the matrix defined as:

$$\boldsymbol{I}_{\otimes \mathrm{B}} = \sum_i m_i \left[(\boldsymbol{b}_i \bullet \boldsymbol{b}_i) \boldsymbol{I}_{3\times3} - \boldsymbol{b}_i \otimes \boldsymbol{b}_i \right] + \iiint_{\boldsymbol{V}} \rho(\boldsymbol{b}) \left[(\boldsymbol{b} \bullet \boldsymbol{b}) \boldsymbol{I}_{3\times3} - \boldsymbol{b} \otimes \boldsymbol{b} \right] d\boldsymbol{b}$$

This matrix is the *moment of inertia tensor*[5]. The importance of this matrix is that it is time independent and needs to be computed only once and reduces the computation of $\boldsymbol{L}_{\mathrm{B}}(t)$ to nine multiplications and six additions. Appendix D expands $\boldsymbol{I}_{\otimes\mathrm{B}}$ in terms of body coordinate components:

$$\boldsymbol{I}_{\otimes\mathrm{B}} = \begin{pmatrix} I_{11} & I_{12} & I_{13} \\ I_{12} & I_{22} & I_{23} \\ I_{13} & I_{23} & I_{33} \end{pmatrix}$$

where:

$$I_{11} = \sum_i m_i \left(b_{i,2}^2 + b_{i,3}^2\right) + \iiint \rho\left(b_1, b_2, b_3\right)\left(b_2^2 + b_3^2\right) d\boldsymbol{b}$$
$$I_{22} = \sum_i m_i \left(b_{i,1}^2 + b_{i,3}^2\right) + \iiint \rho\left(b_1, b_2, b_3\right)\left(b_1^2 + b_3^2\right) d\boldsymbol{b}$$
$$I_{33} = \sum_i m_i \left(b_{i,1}^2 + b_{i,2}^2\right) + \iiint \rho\left(b_1, b_2, b_3\right)\left(b_1^2 + b_2^2\right) d\boldsymbol{b}$$

$$I_{12} = -\sum_i m_i b_{i,1} b_{i,2} - \iiint \rho\left(b_1, b_2, b_3\right)\left(b_1 b_2\right) d\boldsymbol{b}$$
$$I_{23} = -\sum_i m_i b_{i,2} b_{i,3} - \iiint \rho\left(b_1, b_2, b_3\right)\left(b_2 b_3\right) d\boldsymbol{b}$$
$$I_{13} = -\sum_i m_i b_{i,1} b_{i,3} - \iiint \rho\left(b_1, b_2, b_3\right)\left(b_1 b_3\right) d\boldsymbol{b}$$

$$\boldsymbol{b}_i = \left(b_{i,1}, b_{i,2}, b_{i,3}\right)^{\mathrm{T}}$$
$$\boldsymbol{b} = \left(b_1, b_2, b_3\right)^{\mathrm{T}}$$

The coordinate component values depend on the choice of the basis for the body coordinate system. The only constraint[6] imposed on the basis is that it is orthonormal with origin at the center of mass. The matrix $\boldsymbol{I}_{\otimes\mathrm{B}}$ is symmetric, thus there exists some basis satisfying the constraint which will diagonalize the matrix and put the diagonal elements (the eigenvalues) in increasing order. That is, there exists a choice of basis for which:

---

[5] The expression for $\boldsymbol{I}_{\otimes\mathrm{B}}$ is bilinear in body coordinates and may therefore be regarded as a tensor of order 2.

[6] Some applications impose additional constraints. For example, in the IEEE 1278.1-1995 standard the first axis in the entity coordinates system must point forward.

$$\boldsymbol{I}_{\otimes\mathrm{B}} = \begin{pmatrix} I_{11} & 0 & 0 \\ 0 & I_{22} & 0 \\ 0 & 0 & I_{33} \end{pmatrix}, \text{ and } I_{11} \le I_{22} \le I_{33}.$$

The coordinate axes of this basis are called the *principal axes*[7] and the diagonal elements I_{11}, I_{22}, and I_{33} are called the *principal moments of inertia*. The use of this basis reduces the computation of $\boldsymbol{L}_{\mathrm{B}}(t)$ to three multiplications.

The total rotational momentum in space coordinates is given by:

$$\boldsymbol{L}(t) = \boldsymbol{I}_{\otimes}(t)\boldsymbol{\omega}(t).$$

In space coordinates, the moment of inertia tensor is time dependent and is computed as:

$$\boldsymbol{I}_{\otimes}(t) = \boldsymbol{\Omega}(t)\boldsymbol{I}_{\otimes,\mathrm{B}}\boldsymbol{\Omega}^{\mathsf{T}}(t)$$

If the external force acting on particle P_i is $\boldsymbol{F}_i(t)$ and/or the external force acting on a volume element is $\boldsymbol{f}(\boldsymbol{s},t)$ in the space coordinate system and $\boldsymbol{F}_{i\mathrm{B}}(t)$ and $\boldsymbol{f}_{\mathrm{B}}(\boldsymbol{b},t)$ in the body coordinate system then the total force acting on the body is:

$$\begin{aligned} \boldsymbol{F}(t) &= \sum_i \boldsymbol{F}_i(t) + \iiint_{s\in V} \boldsymbol{f}(\boldsymbol{s},t)\,d\boldsymbol{s} \\ \boldsymbol{F}_{\mathrm{B}}(t) &= \sum_i \boldsymbol{F}_{i\mathrm{B}}(t) + \iiint_{b\in V} \boldsymbol{f}_{\mathrm{B}}(\boldsymbol{b},t)\,d\boldsymbol{b} \end{aligned}.$$

The analogue for external rotational force is *torque* and it is defined as:

$$\begin{aligned} \boldsymbol{\tau}(t) &= \sum_i \left(\boldsymbol{s}_{\mathrm{CM}}(t) - \boldsymbol{s}_i(t)\right) \times \boldsymbol{F}_i(t) + \iiint_V \left(\boldsymbol{s}_{\mathrm{CM}}(t) - \boldsymbol{s}(t)\right) \times \boldsymbol{f}(\boldsymbol{s},t)\,d\boldsymbol{s} \\ \boldsymbol{\tau}_{\mathrm{B}}(t) &= \sum_i \boldsymbol{b}_i \times \boldsymbol{F}_{i\mathrm{B}}(t) + \iiint_V \boldsymbol{b} \times \boldsymbol{f}_{\mathrm{B}}(\boldsymbol{b},t)\,d\boldsymbol{b} \end{aligned}$$

In Newtonian physics the momentum of a particle is preserved unless an external force acts on it in which case the relationship between force and momentum is:

$$\boldsymbol{F}_i(t) = \frac{d}{dt}\boldsymbol{P}_i(t)$$

It follows that the relationship between total force and total momentum is:

---

[7] Not to be confused with principal rotations.

$$\boldsymbol{F}(t) = \frac{d}{dt}\boldsymbol{P}(t) = M\frac{d}{dt}\boldsymbol{v}(t).$$

Similarly for torque we have:

$$\begin{aligned}\boldsymbol{\tau}_{\mathrm{B}}(t) &= \sum \boldsymbol{b}_i \times \boldsymbol{F}_{\mathrm{B}i}(t) + \iiint \boldsymbol{b} \times \boldsymbol{f}_{\mathrm{B}}(\boldsymbol{b},t)\,d\boldsymbol{b} \\ &= \sum \boldsymbol{b}_i \times \frac{d}{dt}\boldsymbol{p}_i(t) + \iiint \boldsymbol{b} \times \frac{d}{dt}\boldsymbol{f}_{\mathrm{B}}(\boldsymbol{b},t)\,d\boldsymbol{b} \\ &= \frac{d}{dt}\Big[\sum \boldsymbol{b}_i \times \boldsymbol{p}_i(t) + \iiint \rho(\boldsymbol{b})\,\boldsymbol{b} \times \big(\boldsymbol{\omega}_{\mathrm{B}}(t) \times \boldsymbol{b}\big)\,d\boldsymbol{b}\Big] \\ &= \frac{d}{dt}\boldsymbol{L}_{\mathrm{B}}(t)\end{aligned}$$

In term of rotational inertia we have:

$$\boldsymbol{\tau}(t) = \frac{d}{dt}\boldsymbol{L}(t) = \boldsymbol{I}_{\otimes}\frac{d}{dt}\boldsymbol{\omega}(t).$$

5 Use cases

5.1 DIS Euler angles

This use case is illustrated with the problem of converting aircraft orientation, as indicated by its onboard inertial system, to Tait-Bryan angles with respect to WGS 84 Geocentric (DIS Euler angles).

Consider an aircraft that at time t_0 is stationary on an airfield. Its inertial system set so that the artificial horizon is level with the ground and the instrument panel compass north points in the direction of local North. For convenience, its location at t_0 shall be called ground zero. The resulting inertial system readouts of roll, pitch and yaw now indicate the orientation of the aircraft with respect to a Local-centric Euclidian Frame with origin at ground zero, and $\boldsymbol{x}$-, $\boldsymbol{y}$- and z-axes pointing local north, east and down respectively. This linear-space frame is denoted by $\mathbf{E}_0$. At some subsequent time, the aircraft taxies to the runway, takes off and maneuvers, and at time t_1 the roll, pitch and yaw are read out. These values need to be converted to DIS Euler angles.

The t_1 roll, pitch and yaw values correspond to the orientation of one linear-space frame with respect to another. One is the entity space (the Euclidean frame used for the aircraft), see Figure 7. The other is the $\mathbf{E}_0$ frame. Thus the roll, pitch and yaw values are the Tait-Byan angles representation of the orientation of the aircraft coordinate fram with respect to the $\mathbf{E}_0$ frame:

$$\boldsymbol{\Omega}(t)_{\textbf{Aircraft}\to\textbf{E0}}.$$

Let **L** denote a range coordinate frame consisting of Local Tangent Frame Eucludian SRF. If we assume that the origin of **L** is near, or the same as, ground zero (the origin of $\mathbf{E}_0$), then $\mathbf{E}_0$ and **L** are related as shown in the table below:

Axis	$\mathbf{E}_0$ coordinate frame	local tangent frame **L**
x	points to local North	points to local East
y	points to local East	points to local North
z	points to local down	points to local up

This relationship is expressed as the orientation of $\mathbf{E}_0$ with respect to **L**:

$$\boldsymbol{\Omega}_{\mathbf{E0}\to\mathbf{L}} = \begin{pmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & -1 \end{pmatrix}.$$

Let **W** denote the WGS 84 geocentric SRF. The orientation of **L** with respect **W** is denoted as:

$$\boldsymbol{\Omega}_{\mathbf{L}\to\mathbf{W}}.$$

The SRM specifies the computation of the matrix representation of this orientation based on the WGS 84 geodetic coordinate for the origin of **L**.

The orientation of the aircraft with respect to **W** is the given by:

$$\boldsymbol{\Omega}(t)_{\mathbf{Aircraft}\to\mathbf{W}} = \boldsymbol{\Omega}_{\mathbf{L}\to\mathbf{W}}\boldsymbol{\Omega}_{\mathbf{E0}\to\mathbf{L}}\boldsymbol{\Omega}(t)_{\mathbf{Aircraft}\to\mathbf{E0}} \text{ at time } t.$$

The DIS Euler roll, pitch and yaw values at time t are then just the Tait-Bryan angles representation of the orientation $\boldsymbol{\Omega}(t)_{\mathbf{Aircraft}\to\mathbf{W}}$.

5.2 Rigid body integration of state

In the notation of section 4.3.1, define the state of a rigid body system at time t as the ensemble of the location of the center of mass, the orientation, and the linear and angular momentums:

$$\mathbb{S}(t) \equiv \left\{ \boldsymbol{s}_{\mathrm{CM}}(t), \boldsymbol{\Omega}(t), \boldsymbol{P}(t), \boldsymbol{L}(t) \right\}$$

The problem is compute $\mathbb{S}\left(t+\Delta t\right)$ for some small time increment Δt given the previous state $\mathbb{S}\left(t\right)$ and system specific functions for total force and torque. In the most general case, force and torque are functions of time and the state variables. That is:

$$\begin{aligned}\boldsymbol{F}\left(t\right) &= \boldsymbol{F}\left(t,\mathbb{S}\left(t\right)\right)\\ \boldsymbol{\tau}\left(t\right) &= \boldsymbol{\tau}\left(t,\mathbb{S}\left(t\right)\right)\end{aligned}$$

This problem has several applications. For example, in distributed simulations, entity states are locally "dead reckoned" in time steps Δt until authoritative data has been distributed. In computer generated animations, object states are integrated in steps of Δt equal to the frame rate of the animation and covering the time interval of a scene from beginning to end.

The integration step is realized by setting each state variable $\boldsymbol{X}\left(t+\Delta t\right)$ to the approximate value $\boldsymbol{X}\left(t\right)+\Delta t\dfrac{d\boldsymbol{X}\left(t\right)}{dt}$. The error of this approximation decreases as Δt approaches zero. In particular $\mathbb{S}\left(t+\Delta t\right)$ may be approximated by setting:

$$\begin{aligned}\boldsymbol{s}_{\text{CM}}\left(t+\Delta t\right) &= \boldsymbol{s}_{\text{CM}}\left(t\right)+\frac{\Delta t}{M}\boldsymbol{P}\left(t\right)\\ \boldsymbol{\Omega}\left(t+\Delta t\right) &= \boldsymbol{\Omega}\left(t\right)+\Delta t\,\boldsymbol{S}_{\boldsymbol{\omega}(t)}\boldsymbol{\Omega}\left(t\right)\\ \boldsymbol{P}\left(t+\Delta t\right) &= \boldsymbol{P}\left(t\right)+\Delta t\,\boldsymbol{F}\left(t\right)\\ \boldsymbol{L}\left(t+\Delta t\right) &= \boldsymbol{L}\left(t\right)+\Delta t\,\boldsymbol{\tau}\left(t\right)\end{aligned}$$

The in the expression for $\boldsymbol{\Omega}$ the value $\boldsymbol{\omega}\left(t\right)$ is required to determine the corresponding skew-symmetric matrix $\boldsymbol{S}_{\boldsymbol{\omega}(t)}$. That requires two auxiliary computations.

$$\begin{aligned}\boldsymbol{I}_{\otimes}^{-1}\left(t\right) &= \boldsymbol{\Omega}\left(t\right)\boldsymbol{I}_{\otimes,\text{B}}^{-1}\boldsymbol{\Omega}^{\mathsf{T}}\left(t\right)\\ \boldsymbol{\omega}\left(t\right) &= \boldsymbol{I}_{\otimes}^{-1}\left(t\right)\boldsymbol{L}\left(t\right)\end{aligned}$$

The computation for $\boldsymbol{\Omega}\left(t+\Delta t\right)$, like the other computations, is approximate. The approximate value may fail the criteria for an orientation matrix ($\det\boldsymbol{\Omega}=1$, and $\boldsymbol{\Omega}^{\mathsf{T}}\boldsymbol{\Omega}=\boldsymbol{I}_{3\times 3}$). This will lead to undesirable results in subsequent iterations, therefore the approximate $\boldsymbol{\Omega}\left(t+\Delta t\right)$ value needs to be adjusted to satisfy the criteria. If the $\boldsymbol{\Omega}$ operator is represented with a matrix, this adjustment can be computationally expensive. This is one reason for the popular use of unit quaternions. A quaternion is adjusted to a unit quaternion simply by dividing by its scalar modulus.

6 References

[SRM]

Bibliography

Appendices

Appendix A – Properties of the vector cross product

Definition:

$$\boldsymbol{u}\times\boldsymbol{v} = \left(u_2v_3 - u_3v_2,\, u_3v_1 - u_1v_3,\, u_1v_2 - u_2v_1\right)^{\mathsf{T}}$$

Note that $\boldsymbol{v}\times\boldsymbol{u} = -\boldsymbol{u}\times\boldsymbol{v}$ so that the cross production is not a commutative operation.

$$\left\|\boldsymbol{u}\times\boldsymbol{v}\right\| = \left\|\boldsymbol{u}\right\|\left\|\boldsymbol{v}\right\|\sin\theta$$

where θ is the angle between the two vectors.

The following identity is called Lagrange's formula:

$$\boldsymbol{u}\times\left(\boldsymbol{v}\times\boldsymbol{w}\right) = \left(\boldsymbol{u}\bullet\boldsymbol{w}\right)\boldsymbol{v} - \left(\boldsymbol{u}\bullet\boldsymbol{v}\right)\boldsymbol{w}$$

The cross product can be computed as a matrix multiplication. For each vector $\boldsymbol{u}$ there corresponds a skew-symmetric matrix $\boldsymbol{S}_{\boldsymbol{u}} = \begin{pmatrix} 0 & -u_3 & u_2 \\ u_3 & 0 & -u_1 \\ -u_2 & u_1 & 0 \end{pmatrix}$ such that $\boldsymbol{u}\times\boldsymbol{v} = \boldsymbol{S}_{\boldsymbol{u}}\boldsymbol{v}$.

A special case of Lagrange's formula is:

$$\boldsymbol{u}\times\left(\boldsymbol{u}\times\boldsymbol{w}\right) = \left(\boldsymbol{u}\bullet\boldsymbol{w}\right)\boldsymbol{u} - \left(\boldsymbol{u}\bullet\boldsymbol{u}\right)\boldsymbol{w}.$$

If $\boldsymbol{w} = \left(w_1, w_2, w_3\right)^{\mathsf{T}}$, $\boldsymbol{u} = \left(u_1, u_2, u_3\right)^{\mathsf{T}}$, then:

$$\left(\boldsymbol{u}\bullet\boldsymbol{w}\right)\boldsymbol{u}=\left(u_1w_1+u_2w_2+u_3w_3\right)\begin{pmatrix}u_1\\u_2\\u_3\end{pmatrix}$$
$$=\begin{pmatrix}u_1u_1w_1+u_1u_2w_2+u_1u_3w_3\\u_2u_1w_1+u_2u_2w_2+u_2u_3w_3\\u_3u_1w_1+u_3u_2w_2+u_3u_3w_3\end{pmatrix}$$
$$=\begin{pmatrix}u_1u_1 & u_1u_2 & u_1u_3\\u_2u_1 & u_2u_2 & u_2u_3\\u_3u_1 & u_3u_2 & u_3u_3\end{pmatrix}\begin{pmatrix}w_1\\w_2\\w_3\end{pmatrix}$$
$$=\left(\boldsymbol{u}\otimes\boldsymbol{u}\right)\boldsymbol{w}$$

It follows that:

$$\boldsymbol{u}\times\left(\boldsymbol{u}\times\boldsymbol{w}\right)=\left(\boldsymbol{u}\otimes\boldsymbol{u}\right)\boldsymbol{w}-\left(\boldsymbol{u}\bullet\boldsymbol{u}\right)\boldsymbol{w}$$
$$=\left[\left(\boldsymbol{u}\otimes\boldsymbol{u}\right)-\left(\boldsymbol{u}\bullet\boldsymbol{u}\right)\boldsymbol{I}_{3\times3}\right]\boldsymbol{w}$$

Since $\boldsymbol{u}\times\boldsymbol{w}=-\boldsymbol{w}\times\boldsymbol{u}$ we also have:

$$\boldsymbol{u}\times\left(\boldsymbol{w}\times\boldsymbol{u}\right)=\left[\left(\boldsymbol{u}\bullet\boldsymbol{u}\right)\boldsymbol{I}_{3\times3}-\boldsymbol{u}\otimes\boldsymbol{u}\right]\boldsymbol{w}$$

Also, substituting $\boldsymbol{u}\times\boldsymbol{v}=\boldsymbol{S_u}\boldsymbol{v}$, we have $\boldsymbol{u}\times\left(\boldsymbol{u}\times\boldsymbol{w}\right)=\boldsymbol{u}\times\left(\boldsymbol{S_u}\boldsymbol{v}\right)=\boldsymbol{S_u}\left(\boldsymbol{S_u}\boldsymbol{v}\right)=\boldsymbol{S}_{\boldsymbol{u}}^2\boldsymbol{v}$. Therefore:

$$\boldsymbol{S}_{\boldsymbol{u}}^2=\left[\left(\boldsymbol{u}\otimes\boldsymbol{u}\right)-\left(\boldsymbol{u}\bullet\boldsymbol{u}\right)\boldsymbol{I}_{3\times3}\right]$$

Appendix B – Derivation of Rodrigues' rotation formula

Let $\boldsymbol{n}$ be a unit vector and θ a rotation angle. The point $\boldsymbol{r}$ is rotated around the axis determined by $\boldsymbol{n}$ through angle θ to the rotated point $\boldsymbol{r}'$. To compute $\boldsymbol{r}'$ in terms of $\boldsymbol{n}, \theta,$ and $\boldsymbol{r}'$, consider first the special case of a point $\boldsymbol{s}$ that is perpendicular to $\boldsymbol{n}$. A unit vector $\boldsymbol{m}$ that is perpendicular to both $\boldsymbol{s}$ and $\boldsymbol{n}$ is given by:

$$\boldsymbol{m}=\frac{1}{\left\|\boldsymbol{n}\times\boldsymbol{s}\right\|}\boldsymbol{n}\times\boldsymbol{s}=\frac{1}{\left\|\boldsymbol{s}\right\|}\boldsymbol{n}\times\boldsymbol{s}\ \text{ since}\left\|\boldsymbol{n}\times\boldsymbol{s}\right\|=\sin\left(\pi/2\right)\left\|\boldsymbol{n}\right\|\left\|\boldsymbol{s}\right\|=\left\|\boldsymbol{s}\right\|.$$

The point $\boldsymbol{s}$ rotates to the point $\boldsymbol{s}'$ in the plane spanned by $\boldsymbol{m}$ and $\boldsymbol{s}$. The right triangle in Figure B.1 has a hypotenuse of length $\left\|\boldsymbol{s}\right\|$ and sides of lengths $\sin\left(\theta\right)\left\|\boldsymbol{s}\right\|$ and $\cos\left(\theta\right)\left\|\boldsymbol{s}\right\|$. It follows that

$$s' = \cos(\theta)\, s + \sin(\theta)\|s\|\, m = \cos(\theta)\, s + \sin(\theta)\, n \times s \,.$$

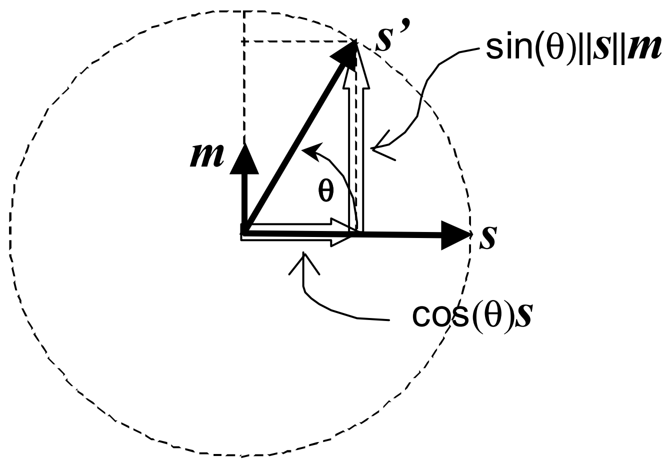

Figure B.1

In the general case, let $s = r - (r \bullet n)\, n$. Then, as shown in Figure B.2, r is the vector sum s and its component in the vector n direction: $r = (r \bullet n)\, n + s$. Since $(r \bullet n)\, n$ is on the n axis, it does not change under the rotation and so $r' = (r \bullet n)\, n + s'$. Substituting $s' = \cos(\theta)\, s + \sin(\theta)\, n \times s$ gives: $r' = (r \bullet n)\, n + \cos(\theta)\, s + \sin(\theta)\, n \times s$. Substitute $n \times s = n \times (r - (r \bullet n)\, n) = n \times r - (r \bullet n)\, n \times n = n \times r$, and substitute $s = r - (r \bullet n)\, n$ to get $r' = (r \bullet n)\, n + \cos(\theta)(r - (r \bullet n)\, n) + \sin(\theta)\, n \times r$. Simplifying the last result we have Rodrigues' rotation formula:

$$r' = \cos(\theta)\, r + (1 - \cos(\theta))(r \bullet n)\, n + \sin(\theta)\, n \times r$$

Matrix form: As a consequence of Lagrange's formula, $n \times (n \times r) = (n \bullet r)\, n - (n \bullet n)\, r$ and since n is a unit vector, $(n \bullet n) = 1$ and we have $(r \bullet n)\, n = n \times (n \times r) + r$. Substituting for this term in Rodrigues' rotation formula yields the following alternate form:

$$\begin{aligned} r' &= \cos(\theta)\, r + (1 - \cos(\theta))\left[n \times (n \times r) + r \right] + \sin(\theta)\, n \times r \\ &= r + (1 - \cos(\theta))\, n \times (n \times r) + \sin(\theta)\, n \times r \end{aligned}$$

Substituting with the matrix form of the cross product gives the matrix form of the formula:

$\boldsymbol{r}' = \left[\boldsymbol{I}_{3\times3} + \left(1-\cos(\theta)\right)\boldsymbol{S}_{\boldsymbol{n}}^2 + \sin(\theta)\boldsymbol{S}_{\boldsymbol{n}} \right]\boldsymbol{r}$, where $\boldsymbol{S}_{\boldsymbol{n}} = \begin{pmatrix} 0 & -n_3 & n_2 \\ n_3 & 0 & -n_1 \\ -n_2 & n_1 & 0 \end{pmatrix}$ is the skew matrix corresponding to $\boldsymbol{n}$.

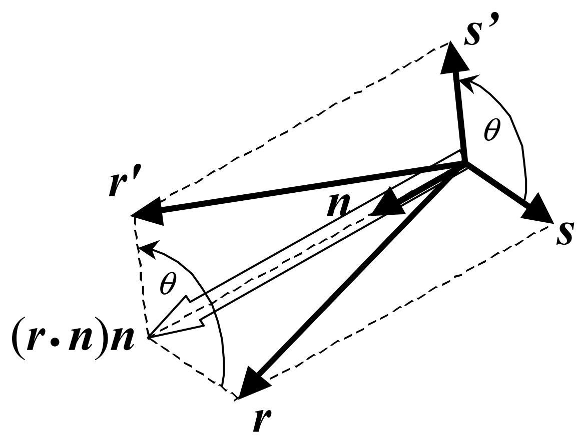

Figure B.2

Appendix C – Quaternion operators on 3D Euclidean space derivation

This appendix provides the derivation of the equality:

$$\boldsymbol{q}(0,\boldsymbol{r})\boldsymbol{q}^* = \left(0,\ \left(\left(e_0^2 - \boldsymbol{e}\bullet\boldsymbol{e}\right)\boldsymbol{r} + 2\left(\boldsymbol{e}\bullet\boldsymbol{r}\right)\boldsymbol{e} + 2e_0\boldsymbol{e}\times\boldsymbol{r}\right)\right)$$

which shows that the quaternion operation $\boldsymbol{p} \mapsto \boldsymbol{q}\boldsymbol{p}\boldsymbol{q}^*$ transforms a "pure imaginary" quaternion to another "pure imaginary" quaternion and that the imaginary part $\boldsymbol{r}$ is transformed to $\left(e_0^2 - \boldsymbol{e}\bullet\boldsymbol{e}\right)\boldsymbol{r} + 2\left(\boldsymbol{e}\bullet\boldsymbol{r}\right)\boldsymbol{e} + 2e_0\boldsymbol{e}\times\boldsymbol{r}$ where $\boldsymbol{q} = \left(e_0, \boldsymbol{e}\right)$.

Substitute $\boldsymbol{q} = \left(e_0, \boldsymbol{e}\right)$:

$$\boldsymbol{q}(0,\boldsymbol{r})\boldsymbol{q}^{*}=(e_0,\boldsymbol{e})(0,\boldsymbol{r})(e_0,-\boldsymbol{e})$$

left multiply:

$$=(0-\boldsymbol{e}\bullet\boldsymbol{r},\ \ e_0\boldsymbol{r}+\boldsymbol{e}\times\boldsymbol{r})(e_0,-\boldsymbol{e})$$

right multiply:

$$=\begin{pmatrix}-e_0\boldsymbol{e}\bullet\boldsymbol{r}-(e_0\boldsymbol{r}+\boldsymbol{e}\times\boldsymbol{r})\bullet(-\boldsymbol{e}), \\ \left(e_0(e_0\boldsymbol{r}+\boldsymbol{e}\times\boldsymbol{r})+(-\boldsymbol{e}\bullet\boldsymbol{r})(-\boldsymbol{e})+(e_0\boldsymbol{r}+\boldsymbol{e}\times\boldsymbol{r})\times(-\boldsymbol{e})\right)\end{pmatrix}$$

$$=\begin{pmatrix}-e_0\boldsymbol{e}\bullet\boldsymbol{r}+(e_0\boldsymbol{r}+\boldsymbol{e}\times\boldsymbol{r})\bullet\boldsymbol{e}, \\ \left(e_0(e_0\boldsymbol{r}+\boldsymbol{e}\times\boldsymbol{r})+(\boldsymbol{e}\bullet\boldsymbol{r})\boldsymbol{e}-(e_0\boldsymbol{r}+\boldsymbol{e}\times\boldsymbol{r})\times\boldsymbol{e}\right)\end{pmatrix}$$

distribute terms:

$$=\begin{pmatrix}-e_0\boldsymbol{e}\bullet\boldsymbol{r}+e_0\boldsymbol{r}\bullet\boldsymbol{e}+(\boldsymbol{e}\times\boldsymbol{r})\bullet\boldsymbol{e}, \\ \left((e_0e_0\boldsymbol{r}+e_0\boldsymbol{e}\times\boldsymbol{r})+(\boldsymbol{e}\bullet\boldsymbol{r})\boldsymbol{e}-e_0\boldsymbol{r}\times\boldsymbol{e}-(\boldsymbol{e}\times\boldsymbol{r})\times\boldsymbol{e}\right)\end{pmatrix}$$

simplify:

$$=\left((\boldsymbol{e}\times\boldsymbol{r})\bullet\boldsymbol{e},\ \left(e_0^2\boldsymbol{r}+e_0\boldsymbol{e}\times\boldsymbol{r}+(\boldsymbol{e}\bullet\boldsymbol{r})\boldsymbol{e}-e_0\boldsymbol{r}\times\boldsymbol{e}-(\boldsymbol{e}\times\boldsymbol{r})\times\boldsymbol{e}\right)\right)$$

since: $(\boldsymbol{e}\times\boldsymbol{r})$ is perpendicular to $\boldsymbol{e}$, $(\boldsymbol{e}\times\boldsymbol{r})\bullet\boldsymbol{e}=0$,

and: $-(\boldsymbol{e}\times\boldsymbol{r})\times\boldsymbol{e}=\boldsymbol{e}\times(\boldsymbol{e}\times\boldsymbol{r})=+(\boldsymbol{e}\bullet\boldsymbol{r})\boldsymbol{e}-(\boldsymbol{e}\bullet\boldsymbol{e})\boldsymbol{r}$ [Lagrange's formula]

$$=\left(0,\ \left(e_0^2\boldsymbol{r}+e_0\boldsymbol{e}\times\boldsymbol{r}+(\boldsymbol{e}\bullet\boldsymbol{r})\boldsymbol{e}-e_0\boldsymbol{r}\times\boldsymbol{e}+(\boldsymbol{e}\bullet\boldsymbol{r})\boldsymbol{e}-(\boldsymbol{e}\bullet\boldsymbol{e})\boldsymbol{r}\right)\right)$$

since: $-e_0\boldsymbol{r}\times\boldsymbol{e}=+e_0\boldsymbol{e}\times\boldsymbol{r}$

$$=\left(0,\ \left(e_0^2\boldsymbol{r}+(\boldsymbol{e}\bullet\boldsymbol{r})\boldsymbol{e}+2e_0\boldsymbol{e}\times\boldsymbol{r}+(\boldsymbol{e}\bullet\boldsymbol{r})\boldsymbol{e}-(\boldsymbol{e}\bullet\boldsymbol{e})\boldsymbol{r}\right)\right)$$

simplify:

$$=\left(0,\ \left((e_0^2-\boldsymbol{e}\bullet\boldsymbol{e})\boldsymbol{r}+2(\boldsymbol{e}\bullet\boldsymbol{r})\boldsymbol{e}+2e_0\boldsymbol{e}\times\boldsymbol{r}\right)\right)$$

Appendix D – Moment of inertia

The definition of total angular momentum $\boldsymbol{L}$ is a summation and/or integration of terms in form $\boldsymbol{b}\times(\boldsymbol{\omega}\times\boldsymbol{b})$. As shown in Appendix A:

$$\boldsymbol{b}\times(\boldsymbol{\omega}\times\boldsymbol{b})=\left[(\boldsymbol{b}\bullet\boldsymbol{b})\boldsymbol{I}_{3\times3}-(\boldsymbol{b}\otimes\boldsymbol{b})\right]\boldsymbol{\omega}.$$

Substituting for this expression in the definition of total momentum yields:

$$\boldsymbol{L}(t) = \left\{ \sum_i m_i \left[\left(\boldsymbol{b}_i \bullet \boldsymbol{b}_i \right) \boldsymbol{I}_{3\times 3} - \boldsymbol{b}_i \otimes \boldsymbol{b}_i \right] + \iiint_{\mathrm{V}} \rho(\boldsymbol{b}) \left[\left(\boldsymbol{b} \bullet \boldsymbol{b} \right) \boldsymbol{I}_{3\times 3} - \boldsymbol{b} \otimes \boldsymbol{b} \right] d\boldsymbol{b} \right\} \boldsymbol{\omega}$$
$$= \boldsymbol{I}_{\otimes} \boldsymbol{\omega}$$

Expressed in coordinate components, the expression $\left[\left(\boldsymbol{b} \bullet \boldsymbol{b} \right) \boldsymbol{I}_{3\times 3} - \left(\boldsymbol{b} \otimes \boldsymbol{b} \right) \right]$ is:

$$\left[\left(\boldsymbol{b} \bullet \boldsymbol{b} \right) \boldsymbol{I}_{3\times 3} - \left(\boldsymbol{b} \otimes \boldsymbol{b} \right) \right] = \left[\begin{pmatrix} b_1^2 + b_2^2 + b_3^2 & 0 & 0 \\ 0 & b_1^2 + b_2^2 + b_3^2 & 0 \\ 0 & 0 & b_1^2 + b_2^2 + b_3^2 \end{pmatrix} - \begin{pmatrix} b_1^2 & b_1 b_2 & b_1 b_3 \\ b_2 b_1 & b_2^2 & b_2 b_3 \\ b_3 b_1 & b_3 b_2 & b_3^2 \end{pmatrix} \right]$$
$$= \begin{pmatrix} b_2^2 + b_3^2 & -b_1 b_2 & -b_1 b_3 \\ -b_2 b_1 & b_1^2 + b_3^2 & -b_2 b_3 \\ -b_3 b_1 & -b_3 b_2 & b_1^2 + b_2^2 \end{pmatrix}$$

Appendix E – Matrix to axis-angle derivation

If $\boldsymbol{R} = \begin{pmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{pmatrix}$ is a rotation matrix, then

$$\boldsymbol{R} = \left[\cos(\theta) \boldsymbol{I}_{3\times 3} + \left(1 - \cos(\theta) \right) \boldsymbol{n} \otimes \boldsymbol{n} + \sin(\theta) \boldsymbol{S}_{\boldsymbol{n}} \right]$$

for some unit vector and angle $(\boldsymbol{n}, \theta)$ (see 3.2.1.1, alternate matrix form of Rodrigues' rotation formula).

The transpose operator is linear and since $\boldsymbol{I}_{3\times 3}$ and $\boldsymbol{n} \otimes \boldsymbol{n}$ are symmetric and $\boldsymbol{S}_{\boldsymbol{n}}$ is skew-symmetric, it follows that:

$$\boldsymbol{R} - \boldsymbol{R}^{\mathrm{T}} = 2 \sin(\theta) \boldsymbol{S}_{\boldsymbol{n}}.$$

The trace operator is also linear and since $\mathrm{Trace}(\boldsymbol{S}_{\boldsymbol{n}}) = 0$ and $\mathrm{Trace}(\boldsymbol{n} \otimes \boldsymbol{n}) = 1$,

$$\mathrm{Trace}(\boldsymbol{R}) = \cos(\theta)(3) + \left(1 - \cos(\theta) \right)(1) + \sin(\theta)(0) = 1 + 2\cos(\theta).$$

Therefore $\theta = \arccos\left(\left(\frac{1 - \mathrm{Trace}(\boldsymbol{R})}{2} \right) \right),\ 0 \le \theta \le \pi.$

If $0 < \theta < \pi$, then $\sin\theta > 0$ and

$$\boldsymbol{S_n} = \begin{pmatrix} 0 & -n_3 & n_2 \\ n_3 & 0 & -n_1 \\ -n_2 & n_1 & 0 \end{pmatrix} = \frac{1}{2\sin(\theta)}\left[\boldsymbol{R} - \boldsymbol{R}^{\mathrm{T}}\right]$$

$$= \frac{1}{2\sin(\theta)}\begin{pmatrix} 0 & a_{12}-a_{21} & a_{13}-a_{31} \\ a_{21}-a_{12} & 0 & a_{23}-a_{32} \\ a_{31}-a_{13} & a_{32}-a_{23} & 0 \end{pmatrix}$$

Therefore:

$$\boldsymbol{n} = \begin{pmatrix} n_1 \\ n_2 \\ n_3 \end{pmatrix} = \frac{1}{2\sin(\theta)}\begin{pmatrix} a_{32}-a_{23} \\ a_{13}-a_{31} \\ a_{21}-a_{12} \end{pmatrix}.$$

Alternatively, let $\boldsymbol{v} = \begin{pmatrix} a_{32}-a_{23} \\ a_{13}-a_{31} \\ a_{21}-a_{12} \end{pmatrix}$, and let $\boldsymbol{n} = \frac{1}{\|\boldsymbol{v}\|}\boldsymbol{v}$.

If $\theta = 0$, there is no rotation so $\boldsymbol{n}$ is in determinant.

If $\theta = \pi$, then the alternate matrix form of Rodrigues' rotation formula reduces down to:

$$\boldsymbol{R} = \left[-\boldsymbol{I}_{3\times 3} + 2\boldsymbol{n}\otimes\boldsymbol{n}\right]$$

$$= \begin{pmatrix} 2n_1^2-1 & 2n_1n_2 & 2n_1n_3 \\ 2n_2n_1 & 2n_2^2-1 & 2n_2n_3 \\ 2n_3n_1 & 2n_3n_2 & 2n_3^2-1 \end{pmatrix}$$

Therefore: $n_1^2 = (a_{11}+1)/2$, $n_2^2 = (a_{22}+1)/2$, and $n_3^2 = (a_{33}+1)/2$. Find the maximum of a_{11}, a_{22}, and a_{33} and use it to find one coordinate component of $\boldsymbol{n}$ and then derive the components from it.

Case a_{11} is the maximum: Let $n_1 = \sqrt{(a_{11}+1)/2}$, $n_2 = \dfrac{a_{12}}{2n_1}$, $n_3 = \dfrac{a_{13}}{2n_1}$.

Case a_{22} is the maximum: Let $n_2 = \sqrt{(a_{22}+1)/2}$, $n_1 = \dfrac{a_{12}}{2n_2}$, $n_3 = \dfrac{a_{32}}{2n_2}$.

Case a_{33} is the maximum: Let $n_3 = \sqrt{(a_{33}+1)/2}$, $n_1 = \dfrac{a_{13}}{2n_3}$, $n_2 = \dfrac{a_{23}}{2n_3}$.

Computational round off may require normalization: $\boldsymbol{n} \to \dfrac{1}{\|\boldsymbol{n}\|}\boldsymbol{n}$.

INDEX