Quaternions & Rotation in 3D Space

Chapter 7-A1

Overview

- Quaternions: definition
- Quaternion properties
- Quaternions and rotation matrices
- Quaternion-rotation matrices relationship
- Spherical linear interpolation
- Concluding remarks

Quaternions

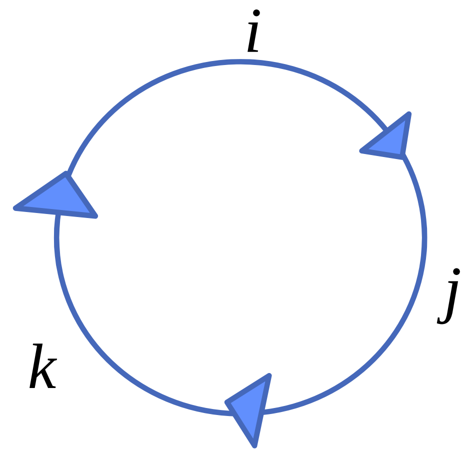
$$\boldsymbol{q} = \underbrace{\boldsymbol{q_o}}_{\textbf{Real Part}} + \underbrace{\boldsymbol{q_x i} + \boldsymbol{q_y j} + \boldsymbol{q_z k}}_{\textbf{Imaginary Part}}$$

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

$$i = jk = -kj$$

$$j = ki = -ik$$

$$k = ij = -ji$$



- The real part for a “Pure Quaternion” is zero.

Quaternion Multiplication

$$\boldsymbol{q_a = q_{o_a} + q_{x_a} i + q_{y_a} j + q_{z_a} k = q_{o_a} + \vec{q}_a = (q_{o_a}; \vec{q}_a)}$$

$$\boldsymbol{q_b = q_{o_b} + q_{x_b} i + q_{y_b} j + q_{z_b} k = q_{o_b} + \vec{q}_b = (q_{o_b}; \vec{q}_b)}$$

$$\begin{aligned}
&\boldsymbol{q_a q_b} \\
&\boldsymbol{= q_{o_a} \left(q_{o_b} + q_{x_b} i + q_{y_b} j + q_{z_b} k\right)} \\
&\boldsymbol{+ q_{x_a} i \left(q_{o_b} + q_{x_b} i + q_{y_b} j + q_{z_b} k\right)} \\
&\boldsymbol{+ q_{y_a} j \left(q_{o_b} + q_{x_b} i + q_{y_b} j + q_{z_b} k\right)} \\
&\boldsymbol{+ q_{z_a} k \left(q_{o_b} + q_{x_b} i + q_{y_b} j + q_{z_b} k\right)}
\end{aligned}$$

- Using the rules in the previous slide, we can get the following definition for quaternion multiplication:

$$\boldsymbol{q_a q_b = (q_{o_a} q_{o_b} - \vec{q}_a . \vec{q}_b; q_{o_a} \vec{q}_b + q_{o_b} \vec{q}_a + \vec{q}_a \times \vec{q}_b)}$$

Quaternion Multiplication

$$q_a = q_{o_a} + q_{x_a} i + q_{y_a} j + q_{z_a} k = q_{o_a} + \vec{q}_a = (q_{o_a}; \vec{q}_a)$$

$$q_b = q_{o_b} + q_{x_b} i + q_{y_b} j + q_{z_b} k = q_{o_b} + \vec{q}_b = (q_{o_b}; \vec{q}_b)$$

$$q_a q_b = C_{q_a} q_b = \bar{C}_{q_b} q_a$$

$$C_{q_a} = \begin{bmatrix} q_{o_a} & -q_{x_a} & -q_{y_a} & -q_{z_a} \\ q_{x_a} & q_{o_a} & -q_{z_a} & q_{y_a} \\ q_{y_a} & q_{z_a} & q_{o_a} & -q_{x_a} \\ q_{z_a} & -q_{y_a} & q_{x_a} & q_{o_a} \end{bmatrix}$$

$$\bar{C}_{q_b} = \begin{bmatrix} q_{o_b} & -q_{x_b} & -q_{y_b} & -q_{z_b} \\ q_{x_b} & q_{o_b} & q_{z_b} & -q_{y_b} \\ q_{y_b} & -q_{z_b} & q_{o_b} & q_{x_b} \\ q_{z_b} & q_{y_b} & -q_{x_b} & q_{o_b} \end{bmatrix}$$

- C_{q_a} & $\bar{C}_{q_b}$ simplify the quaternion multiplication to matrix multiplication – ortho-normal matrices.

Quaternion Multiplication

- Unit quaternions:

$$q_o^2 + q_x^2 + q_y^2 + q_z^2 = 1$$

  - For unit quaternions:

$$\boldsymbol{C_{q_a} C_{q_a}^T} = \begin{bmatrix} \boldsymbol{q_{o_a}} & \boldsymbol{-q_{x_a}} & \boldsymbol{-q_{y_a}} & \boldsymbol{-q_{z_a}} \\ \boldsymbol{q_{x_a}} & \boldsymbol{q_{o_a}} & \boldsymbol{-q_{z_a}} & \boldsymbol{q_{y_a}} \\ \boldsymbol{q_{y_a}} & \boldsymbol{q_{z_a}} & \boldsymbol{q_{o_a}} & \boldsymbol{-q_{x_a}} \\ \boldsymbol{q_{z_a}} & \boldsymbol{-q_{y_a}} & \boldsymbol{q_{x_a}} & \boldsymbol{q_{o_a}} \end{bmatrix} \begin{bmatrix} \boldsymbol{q_{o_a}} & \boldsymbol{q_{x_a}} & \boldsymbol{q_{y_a}} & \boldsymbol{q_{z_a}} \\ \boldsymbol{-q_{x_a}} & \boldsymbol{q_{o_a}} & \boldsymbol{q_{z_a}} & \boldsymbol{-q_{y_a}} \\ \boldsymbol{-q_{y_a}} & \boldsymbol{-q_{z_a}} & \boldsymbol{q_{o_a}} & \boldsymbol{q_{x_a}} \\ \boldsymbol{-q_{z_a}} & \boldsymbol{q_{y_a}} & \boldsymbol{-q_{x_a}} & \boldsymbol{q_{o_a}} \end{bmatrix} = \boldsymbol{I_4}$$

$$\boldsymbol{\overline{C}_{q_b} \overline{C}_{q_b}^T = I_4}$$

Quaternion Properties

- Quaternion conjugate:

$$\boldsymbol{q_a = q_{o_a} + q_{x_a} i + q_{y_a} j + q_{z_a} k = q_{o_a} + \vec{q}_a = (q_{o_a}; \vec{q}_a)}$$

$$\boldsymbol{q_a^* = q_{o_a} - q_{x_a} i - q_{y_a} j - q_{z_a} k = q_{o_a} - \vec{q}_a = (q_{o_a}; -\vec{q}_a)}$$

$$\boldsymbol{q_a q_a^* = (q_{o_a} + \vec{q}_a)(q_{o_a} - \vec{q}_a)}$$

$$\boldsymbol{q_a q_a^* = (q_{o_a}^2 + \vec{q}_a . \vec{q}_a;\ q_{o_a}\vec{q}_a - q_{o_a}\vec{q}_a + \vec{q}_a \times \vec{q}_a)}$$

  - For unit quaternions:

$$\boldsymbol{q_a q_a^* = (1; 0)}$$

Quaternion Properties

- Quaternion conjugate:

$$\boldsymbol{C}_{q_a^*} = \begin{bmatrix} q_{o_a} & q_{x_a} & q_{y_a} & q_{z_a} \\ -q_{x_a} & q_{o_a} & q_{z_a} & -q_{y_a} \\ -q_{y_a} & -q_{z_a} & q_{o_a} & q_{x_a} \\ -q_{z_a} & q_{y_a} & -q_{x_a} & q_{o_a} \end{bmatrix} = \boldsymbol{C}_{q_a}^T$$

$$\overline{\boldsymbol{C}}_{q_b^*} = \begin{bmatrix} q_{o_b} & q_{x_b} & q_{y_b} & q_{z_b} \\ -q_{x_b} & q_{o_b} & -q_{z_b} & q_{y_b} \\ -q_{y_b} & q_{z_b} & q_{o_b} & -q_{x_b} \\ -q_{z_b} & -q_{y_b} & q_{x_b} & q_{o_b} \end{bmatrix} = \overline{\boldsymbol{C}}_{q_b}^T$$

. operator is the dot operator between two quaternions

$$q_a.(q_b q_c q_b^*) = q_a.(\overline{C}_{q_b}^* q_b q_c) = ({C^*_{q_b}}^T q_a).(q_b q_c) = (C_{q_b} q_a).(q_b q_c) = (q_a q_b).(q_b q_c)$$

Quaternions & Rotation Matrices

$$\vec{u}\ is\ a\ unit\ vector$$

- Given the following quaternions:

$q = cos\theta + sin\theta\vec{u}$ $\qquad\qquad$ $q^* = cos\theta - sin\theta\vec{u}$

- q is a unit quaternion.
- $a\dot{u}$ is a pure quaternion (real part is zero).

$$a\vec{u} \quad \Longleftrightarrow \quad a\dot{u} = (0; a\vec{u})$$

$$\boldsymbol{qa\dot{u}q^* = aq\dot{u}q^*}$$

$$q\dot{u} = (cos\theta; sin\theta\vec{u})(0; \vec{u}) = (-sin\theta\vec{u}.\vec{u}; cos\theta\vec{u} + sin\theta\vec{u} \times \vec{u})$$

$$q\dot{u} = (-sin\theta; cos\theta\vec{u})$$

Quaternions & Rotation Matrices

$$q\dot{u}q^* = (-sin\theta; cos\theta\vec{u})(cos\theta; -sin\theta\vec{u})$$

$$q\dot{u}q^* = (-sin\theta cos\theta + sin\theta cos\theta\vec{u}.\vec{u}; \\ cos^2\theta\vec{u} + sin^2\theta\vec{u} - sin\theta cos\theta\vec{u}\times\vec{u})$$

$$\boldsymbol{q\dot{u}q^* = (0; \vec{u})}$$

$$\boldsymbol{qa\dot{u}q^* = a\dot{u}}$$

1

- The product $qa\dot{u}q^*$ produces the same vector $a\dot{u}$.

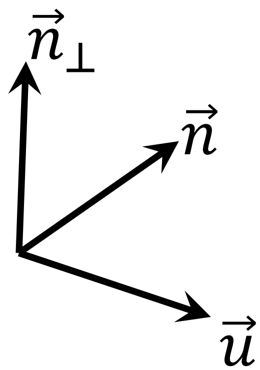
Quaternions & Rotation Matrices

$$\vec{v} \quad \Longleftrightarrow \quad \dot{v} = (0; \vec{n} + a\vec{u})$$

- $\vec{n}$ is perpendicular to $\vec{u}$.

$$q\dot{n} = (cos\theta; sin\theta\vec{u})(0; \vec{n}) = (-sin\theta\vec{u}.\vec{n}; cos\theta\vec{n} + sin\theta\vec{u} \times \vec{n})$$

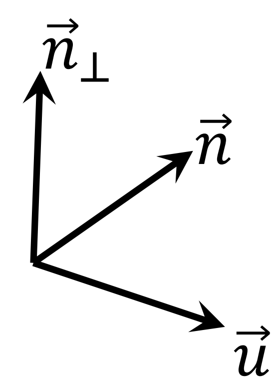
$$q\dot{n} = (cos\theta; sin\theta\vec{u})(0; \vec{n}) = (0; cos\theta\vec{n} + sin\theta\vec{n}_{\perp})$$



Quaternions & Rotation Matrices

$$q\dot{n}q^* = (0; cos\theta\vec{n} + sin\theta\vec{n}_\perp)(cos\theta; -sin\theta\vec{u})$$

$$q\dot{n}q^* \\ = (sin\theta cos\theta\vec{n}.\vec{u} + sin^2\theta\vec{n}_\perp.\vec{u}; cos^2\theta\vec{n} + sin\theta cos\theta\vec{n}_\perp \\ - sin\theta cos\theta\vec{n}\times\vec{u} - sin^2\theta\vec{n}_\perp\times\vec{u})$$



$$q\dot{n}q^* = (0; cos^2\theta\vec{n} + sin\theta cos\theta\vec{n}_\perp + sin\theta cos\theta\vec{n}_\perp - sin^2\theta\vec{n})$$

$$q\dot{n}q^* = (0; [cos^2\theta - sin^2\theta]\vec{n} + 2sin\theta cos\theta\vec{n}_\perp)$$

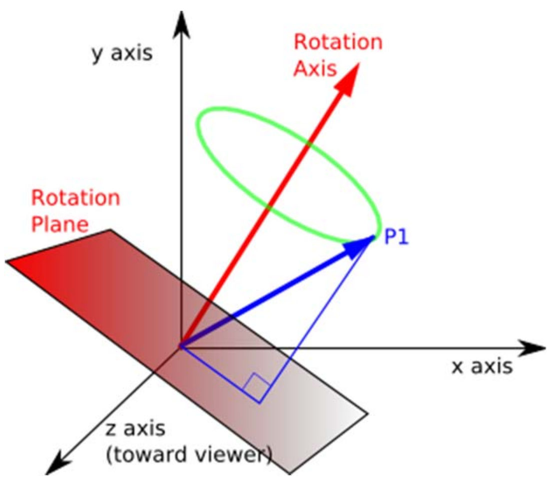
Quaternions & Rotation Matrices

$$q\dot{n}q^* = (0; [cos^2\theta - sin^2\theta]\vec{n} + 2sin\theta cos\theta\vec{n}_\perp)$$

$$\boldsymbol{q\dot{n}q^* = (0; \cos(2\theta)\vec{n} + \sin(2\theta)\vec{n}_\perp)}$$ **2**

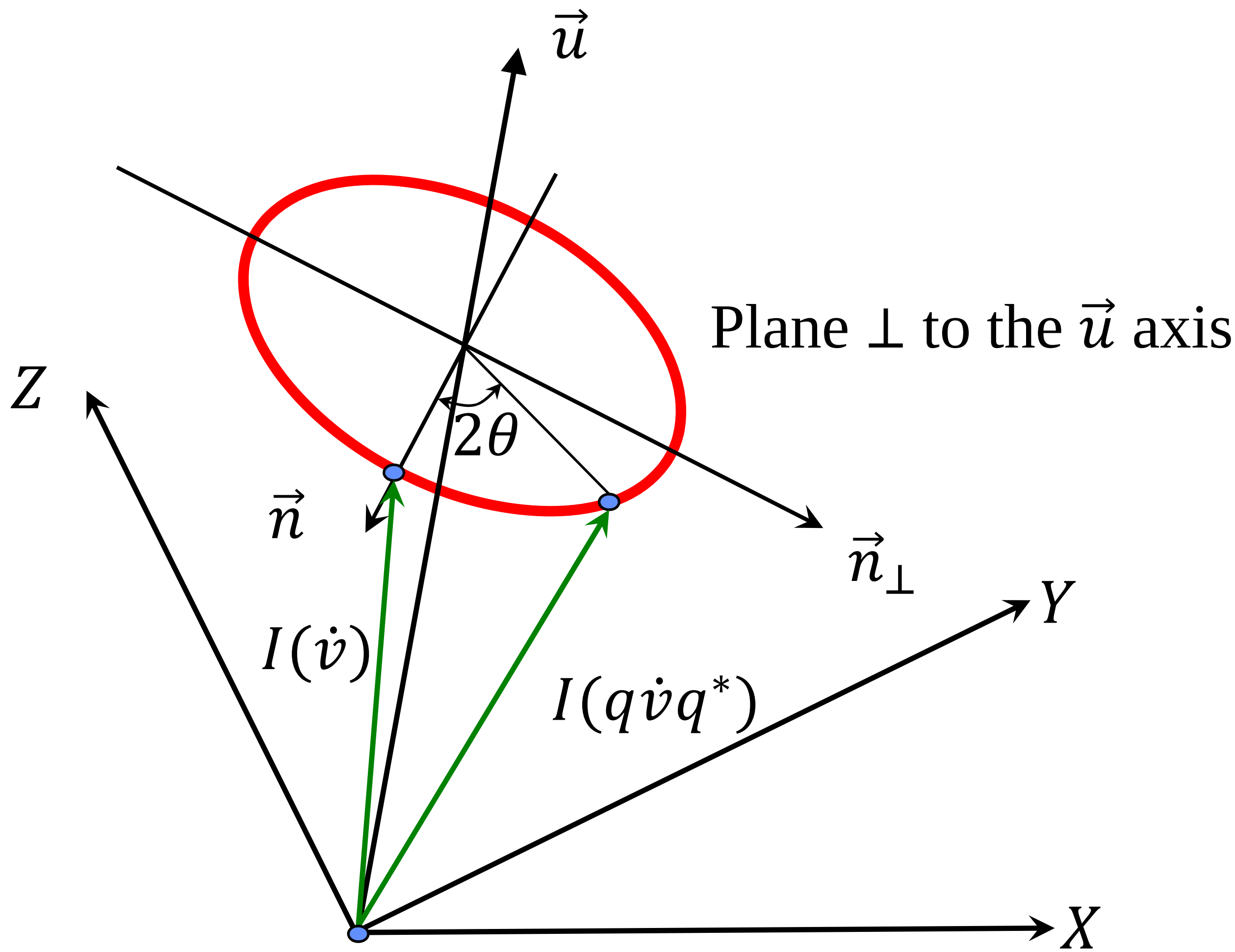
- From 1 & 2, one can conclude that:

$$\boldsymbol{q\dot{v}q^* = q(\dot{n} + a\dot{u})q^* = (0; a\vec{u} + \cos(2\theta)\vec{n} + \sin(2\theta)\vec{n}_\perp)}$$



http://www.euclideanspace.com

Quaternions & Rotation Matrices



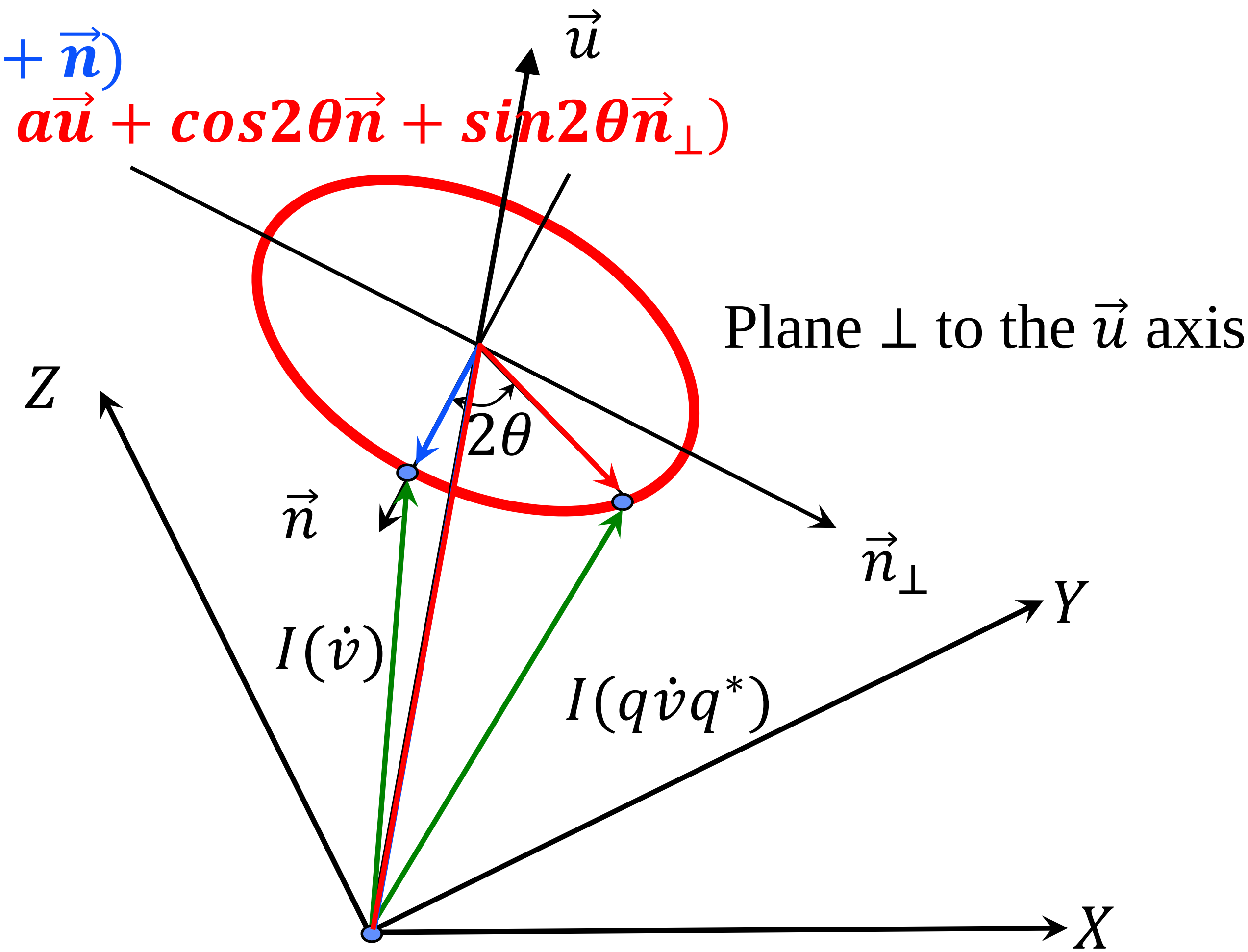
$\dot{v}$ &$q\dot{v}q^*$ are pure quaternions

$I(\dot{v})$ & $I(q\dot{v}q^*)$ are the imaginary components of $\dot{v}$ & $q\dot{v}q^*$.

Quaternions & Rotation Matrices

$\dot{\boldsymbol{v}} = (\mathbf{0}; \boldsymbol{a}\vec{\boldsymbol{u}} + \vec{\boldsymbol{n}})$

$\boldsymbol{q}\dot{\boldsymbol{v}}\boldsymbol{q}^* = (\mathbf{0}; \boldsymbol{a}\vec{\boldsymbol{u}} + \boldsymbol{cos2\theta}\vec{\boldsymbol{n}} + \boldsymbol{sin2\theta}\vec{\boldsymbol{n}}_\perp)$



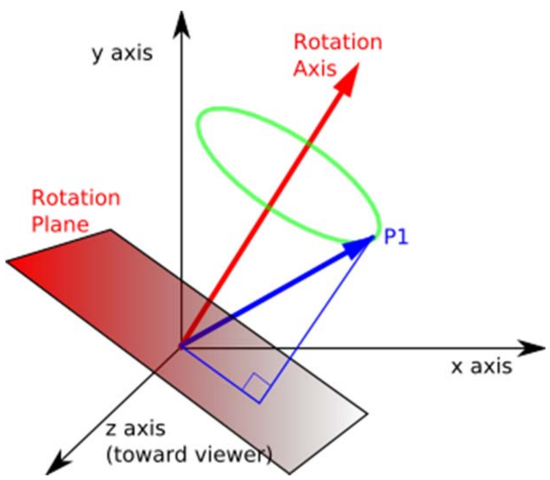
$\dot{v}$ &$q\dot{v}q^*$ are pure quaternions

$I(\dot{v})$ & $I(q\dot{v}q^*)$ are the imaginary components of $\dot{v}$ & $q\dot{v}q^*$.

Quaternions & Rotation Matrices

- Any 3D rotation matrix can be represented by a rotation (θ) around a unit vector ($\vec{u}$).
- This rotation can be defined by the following unit quaternion:

$$q = \cos\left(\frac{\theta}{2}\right) + \sin\left(\frac{\theta}{2}\right) u_x i + \sin\left(\frac{\theta}{2}\right) u_y j + \sin\left(\frac{\theta}{2}\right) u_z k$$



http://www.euclideanspace.com

Quaternions & Rotation Matrices

- Rotation maintains the magnitude of a vector:

$$(q\dot{v}q^*).(q\dot{v}q^*)$$

$$= \left(\bar{C}_{q^*}C_q\dot{v}\right).\left(\bar{C}_{q^*}C_q\dot{v}\right)$$

$$(q\dot{v}q^*).(q\dot{v}q^*) = \dot{v}^T C_q^T \bar{C}_{q^*}^T \bar{C}_{q^*} C_q \dot{v} = \dot{v}^T \dot{v}$$

Quaternions & Rotation Matrices

- Rotation maintains the angular deviation between two vectors:

$$(q\dot{v}_a q^*).(q\dot{v}_b q^*)$$

$$= \left(\bar{C}_{q^*} C_q \dot{v}_a\right).\left(\bar{C}_{q^*} C_q \dot{v}_b\right)$$

$$(q\dot{v}_a q^*).(q\dot{v}_b q^*) = \dot{v}_a^T C_q^T \bar{C}_{q^*}^T \bar{C}_{q^*} C_q \dot{v}_b = \dot{v}_a^T \dot{v}_b$$

Quaternions & Rotation Matrices

- Rotation maintains the magnitude of a triple product:

$$[v_a, v_b, v_c] = v_a . (v_b \times v_c)$$

- Since:
  - Quaternion rotation maintains vector magnitude.
  - Quaternion rotation maintains angular deviation between two vectors.
- Then:
  - Quaternion rotation maintains the magnitude of the triple product.

$$[v_a, v_b, v_c] = [q\dot{v}_a q^*, q\dot{v}_b q^*, q\dot{v}_c q^*]$$

Quaternions & Rotation Matrices

- Quaternion/rotation matrix relationship:

$$R_c^m \vec{v} \Longleftrightarrow q\dot{v}q^*$$

$$q\dot{v}q^* = \bar{C}_{q^*} C_q \dot{v}$$

$$\bar{C}_{q^*} C_q = \begin{bmatrix} q_o & q_x & q_y & q_z \\ -q_x & q_o & -q_z & q_y \\ -q_y & q_z & q_o & -q_x \\ -q_z & -q_y & q_x & q_o \end{bmatrix} \begin{bmatrix} q_o & -q_x & -q_y & -q_z \\ q_x & q_o & -q_z & q_y \\ q_y & q_z & q_o & -q_x \\ q_z & -q_y & q_x & q_o \end{bmatrix}$$

Quaternions & Rotation Matrices

- Quaternion/rotation matrix relationship:

$$\bar{C}_{q^*}C_q = \begin{bmatrix} q_o & q_x & q_y & q_z \\ -q_x & q_o & -q_z & q_y \\ -q_y & q_z & q_o & -q_x \\ -q_z & -q_y & q_x & q_o \end{bmatrix} \begin{bmatrix} q_o & -q_x & -q_y & -q_z \\ q_x & q_o & -q_z & q_y \\ q_y & q_z & q_o & -q_x \\ q_z & -q_y & q_x & q_o \end{bmatrix}$$

$$\bar{C}_{q^*}C_q = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & r_{11} & r_{12} & r_{13} \\ 0 & r_{21} & r_{22} & r_{23} \\ 0 & r_{31} & r_{31} & r_{33} \end{bmatrix}$$

Quaternions & Rotation Matrices

- ***Quaternion to Rotation Transformation***

$$r_{11} = q_x^2 + q_o^2 - q_z^2 - q_y^2$$

$$r_{12} = 2q_x q_y - 2q_o q_z$$

$$r_{13} = 2q_x q_z + 2q_o q_y$$

$$r_{21} = 2q_x q_y + 2q_o q_z$$

$$r_{22} = q_y^2 - q_z^2 + q_o^2 - q_x^2$$

$$r_{23} = 2q_y q_z - 2q_o q_x$$

$$r_{31} = 2q_x q_z - 2q_o q_y$$

$$r_{32} = 2q_y q_z + 2q_o q_x$$

$$r_{33} = q_z^2 - q_y^2 - q_x^2 + q_o^2$$

$\boldsymbol{q}$ & $-\boldsymbol{q}$ define the same rotation matrix

Quaternions & Rotation Matrices

- ***Rotation to Quaternion Transformation (Option # 1)***

$$r_{11} + r_{22} + r_{33} = 3q_o^2 - q_x^2 - q_y^2 - q_z^2$$

$$r_{11} + r_{22} + r_{33} = 4q_o^2 - 1$$

$$\boldsymbol{q_o = \sqrt{(r_{11} + r_{22} + r_{33} + 1)}/2}$$

$$r_{32} - r_{23} = 4q_o q_x$$

$$\boldsymbol{q_x = (r_{32} - r_{23})/4q_o}$$

$$r_{13} - r_{31} = 4q_o q_y$$

$$\boldsymbol{q_y = (r_{13} - r_{31})/4q_o}$$

$$r_{21} - r_{12} = 4q_o q_z$$

$$\boldsymbol{q_z = (r_{21} - r_{12})/4q_o}$$

Assumption: $(r_{11}+r_{22} + r_{33} + 1) > 0$

Quaternions & Rotation Matrices

- ***Rotation to Quaternion Transformation (Option # 2)***

$$r_{11} - r_{22} - r_{33} = 3q_x^2 - q_o^2 - q_y^2 - q_z^2 = 4q_x^2 - 1$$

$$\boldsymbol{q_x = \sqrt{(r_{11} - r_{22} - r_{33} + 1)}/2}$$

$$r_{12} + r_{21} = 4q_x q_y$$

$$\boldsymbol{q_y = (r_{12} + r_{21})/4q_x}$$

$$r_{13} + r_{31} = 4q_x q_z$$

$$\boldsymbol{q_z = (r_{13} + r_{31})/4q_x}$$

$$r_{32} - r_{23} = 4q_o q_x$$

$$\boldsymbol{q_o = (r_{32} - r_{23})/4q_x}$$

Assumption: $(r_{11} - r_{22} - r_{33} + 1) > 0$

Quaternions & Rotation Matrices

- ***Rotation to Quaternion Transformation (Option # 3)***

$$r_{22} - r_{11} - r_{33} = 3q_y^2 - q_o^2 - q_x^2 - q_z^2 = 4q_y^2 - 1$$

$$q_y^2 = (r_{22} - r_{11} - r_{33} + 1)/4$$

$$\mathbf{q_y = \sqrt{(r_{22} - r_{11} - r_{33} + 1)}/2}$$

$$r_{12} + r_{21} = 4q_x q_y$$

$$\mathbf{q_x = (r_{12} + r_{21})/4q_y}$$

$$r_{23} + r_{32} = 4q_y q_z$$

$$\mathbf{q_z = (r_{23} + r_{32})/4q_y}$$

$$r_{13} - r_{31} = 4q_o q_y$$

$$\mathbf{q_o = (r_{13} - r_{31})/4q_y}$$

Assumption: $(r_{22} - r_{11} - r_{33} + 1) > 0$

Quaternions & Rotation Matrices

- ***<u>Rotation to Quaternion Transformation (Option # 4)</u>***

$$r_{33} - r_{11} - r_{22} = 3q_z^2 - q_o^2 - q_x^2 - q_y^2 = 4q_z^2 - 1$$

$$4q_z^2 = (r_{33} - r_{11} - r_{22} + 1)$$

$$\boldsymbol{q_z = \sqrt{(r_{33} - r_{11} - r_{22} + 1)}/2}$$

$$r_{13} + r_{31} = 4q_x q_z$$

$$\boldsymbol{q_x = (r_{13} + r_{31})/4q_z}$$

$$r_{23} + r_{32} = 4q_y q_z$$

$$\boldsymbol{q_y = (r_{23} + r_{32})/4q_z}$$

$$r_{21} - r_{12} = 4q_o q_z$$

$$\boldsymbol{q_o = (r_{21} - r_{12})/4q_z}$$

Assumption: $(r_{33} - r_{11} - r_{22} + 1) > 0$

Quaternions & Rotation Matrices

- ***<u>Rotation to Quaternion Transformation</u>***
- Among the options, choose the one that ensures the highest numerical stability.
- Option # 1: q_o is the largest among $(q_o, q_x, q_y, and\ q_z)$.
- Option # 2: q_x is the largest among $(q_o, q_x, q_y, and\ q_z)$.
- Option # 3: q_y is the largest among $(q_o, q_x, q_y, and\ q_z)$.
- Option # 4: q_z is the largest among $(q_o, q_x, q_y, and\ q_z)$.

Quaternions & Rotation Matrices

Composite Rotations

- The product of two quaternions:

$$q_\alpha = cos\alpha + sin\alpha\,\vec{u} \qquad\qquad q_\beta = cos\beta + sin\beta\,\vec{u}$$

$$q_\alpha q_\beta = (cos\alpha; sin\alpha\,\vec{u})(cos\beta; sin\beta\,\vec{u})$$

$$q_\alpha q_\beta = (cos\alpha\,cos\beta\; - sin\alpha\,sin\beta\,\vec{u}.\vec{u}; \\ cos\alpha\,sin\beta\,\vec{u} + sin\alpha\,cos\beta\,\vec{u} + sin\alpha\,sin\beta\,\vec{u}\times\vec{u})$$

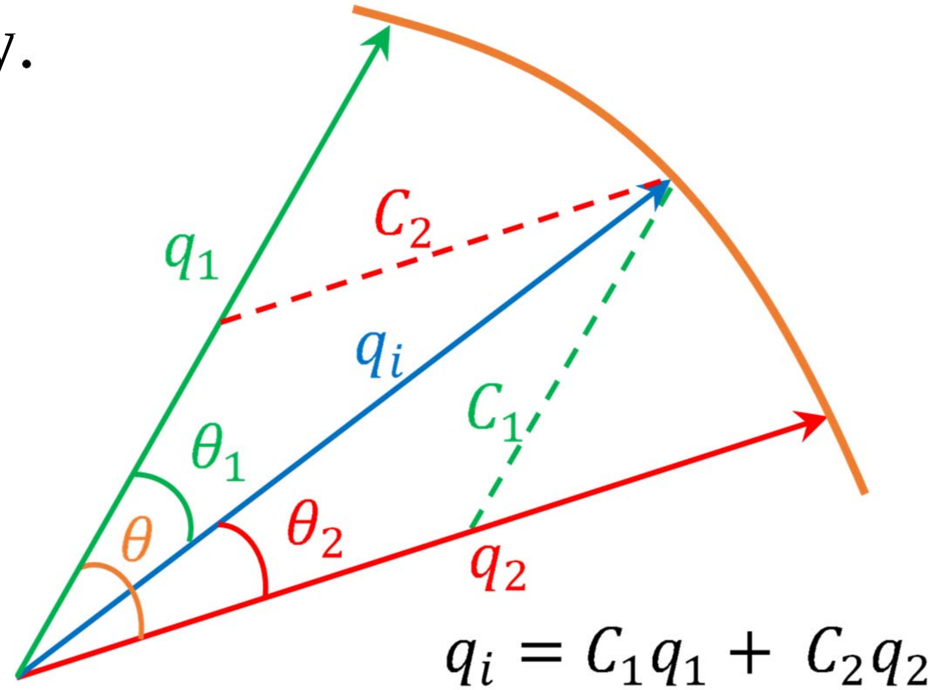
$$q_\alpha q_\beta = (cos\alpha\,cos\beta\; - sin\alpha\,sin\beta; \\ [cos\alpha\,sin\beta + sin\alpha\,cos\beta]\vec{u})$$

$$q_\alpha q_\beta = (\cos[\alpha+\beta]; sin[\alpha+\beta]\vec{u})$$

- This product is equivalent to rotation angle $(2[\alpha+\beta])$ around the axis $\vec{u}$.

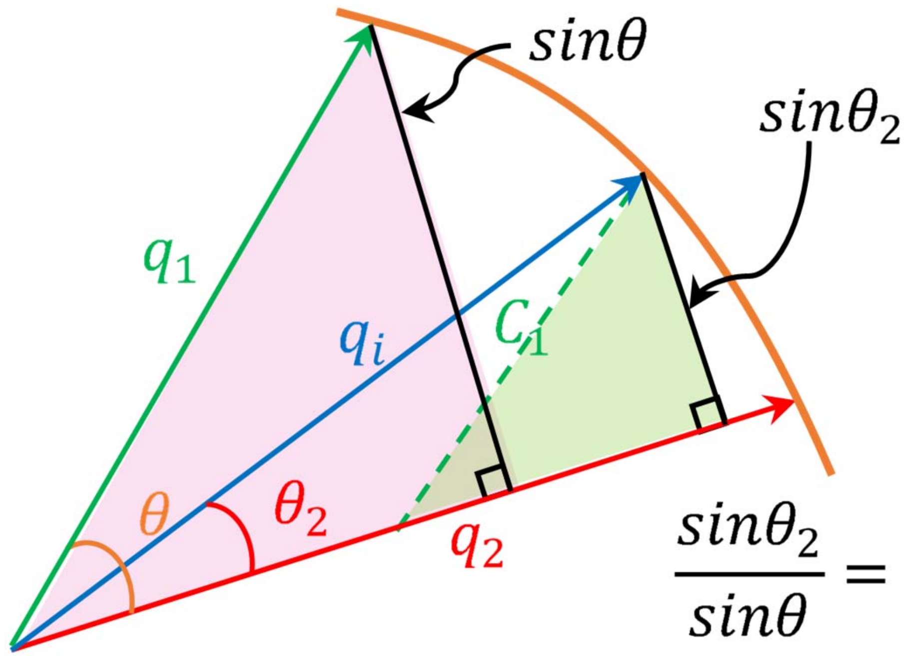
Spherical Linear Interpolation

- **Problem Statement:** Given the rotations represented by q_1 and q_2, whose angular deviation is θ, we need to evaluate the interpolated quaternion rotation q_i, whose angular deviations to q_1 and q_2 are θ_1 and θ_2, respectively.



- As per the figure above: $q_i = C_1 q_1 + C_2 q_2$

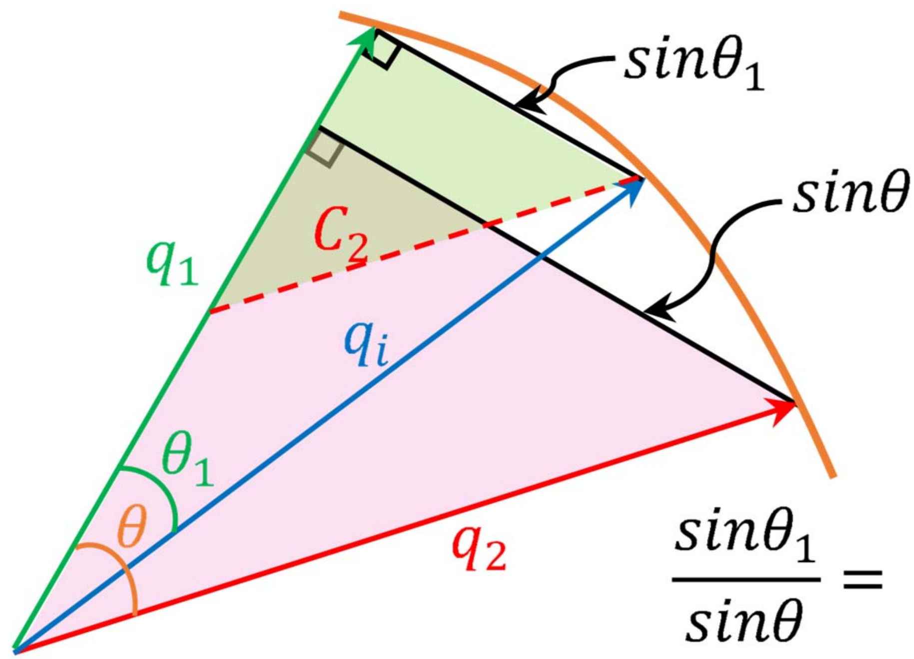
Spherical Linear Interpolation



$$\frac{sin\theta_2}{sin\theta} = \frac{C_1}{1}$$

$$C_1 = \frac{sin\theta_2}{sin\theta}$$

Spherical Linear Interpolation



$$\frac{sin\theta_1}{sin\theta} = \frac{C_2}{1}$$

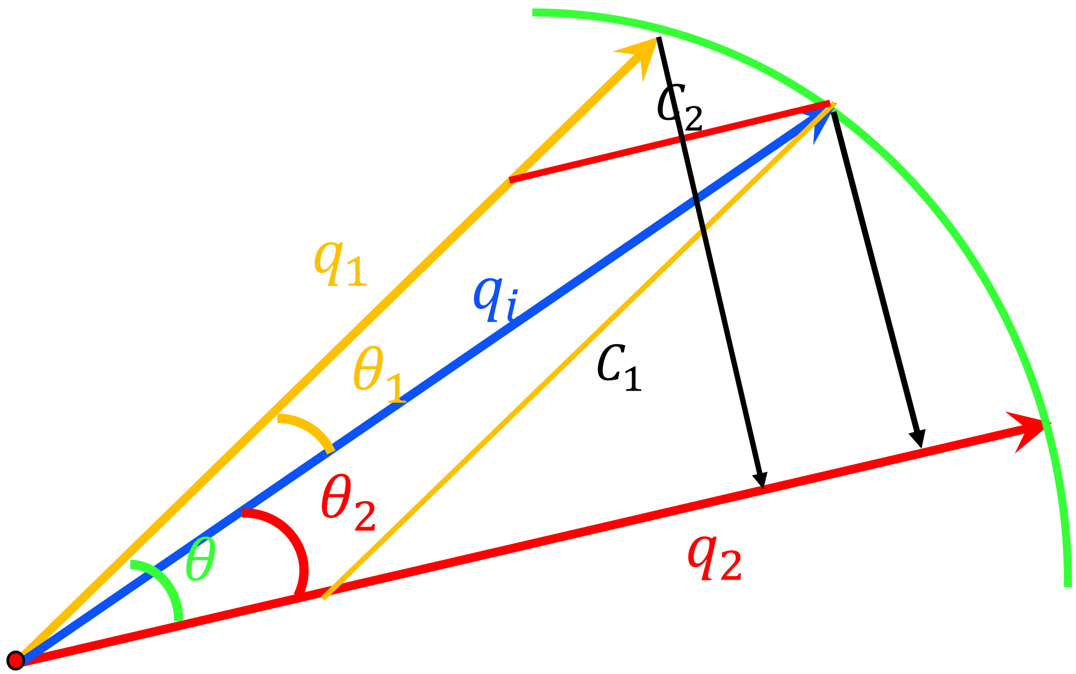
$$C_2 = \frac{sin\theta_1}{sin\theta}$$

Spherical Linear Interpolation

$q_1 . q_2 = cos\theta$

$${}^{C_1}/_{1} = C_1 = {}^{sin\theta_2}/_{sin\theta}$$

$${}^{C_2}/_{1} = C_2 = {}^{sin\theta_1}/_{sin\theta}$$



$$q_i = C_1 q_1 + C_2 q_2$$

$$q_i = {}^{sin\theta_2}/_{sin\theta}\, q_1 + {}^{sin\theta_1}/_{sin\theta}\, q_2$$

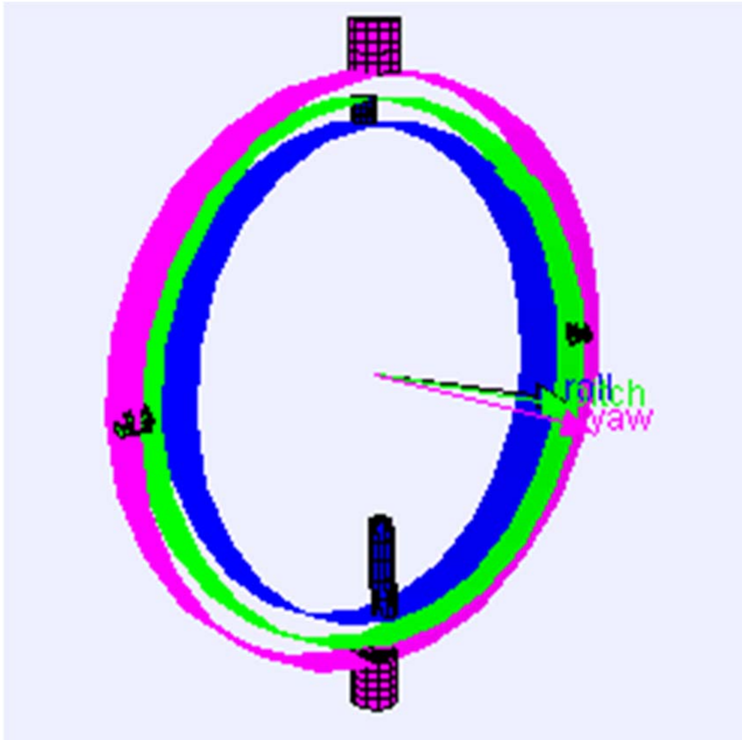
Spherical Linear Interpolation

- Spherical Linear Interpolation is useful for:
  - Interpolation of derived rotation matrices from integrated GNSS/INS attitude – This is the case when deriving the rotation matrices at much higher rate than that derived from GNSS/INS unit (LiDAR & Line Camera systems)
  - Modeling variation of the rotation matrices as time dependent values for Line Camera Systems

Quaternions & Rotation Matrices

- Quaternions characteristics compared to rotation matrices:
  - It avoids the gimbal lock problem.
    - Happens whenever the secondary rotation is 90°
    - Two rotations take place around the same axis in space.
  - Quaternion multiplication requires fewer operations compared to multiplication of two rotation matrices.
  - Quaternion-based rotation requires more operations when compared to traditional rotation of vectors.
  - Quaternions has one constraint while rotation matrices has 6 orthogonality constraints.
  - Interpolation of quaternion rotations is much more straight forward than 3D rotation matrices.

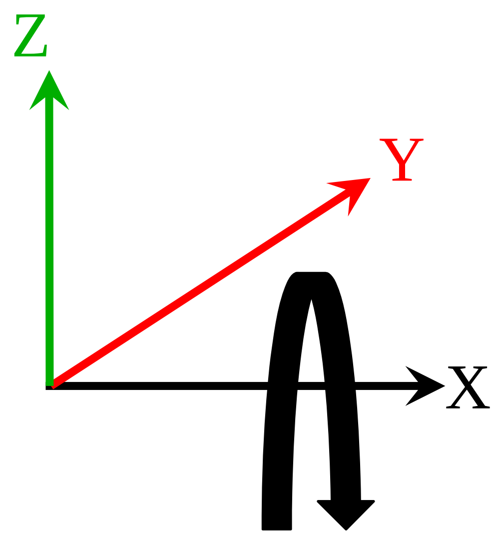
Gimbal Lock



http://en.wikipedia.org/wiki/Gimbal_lock

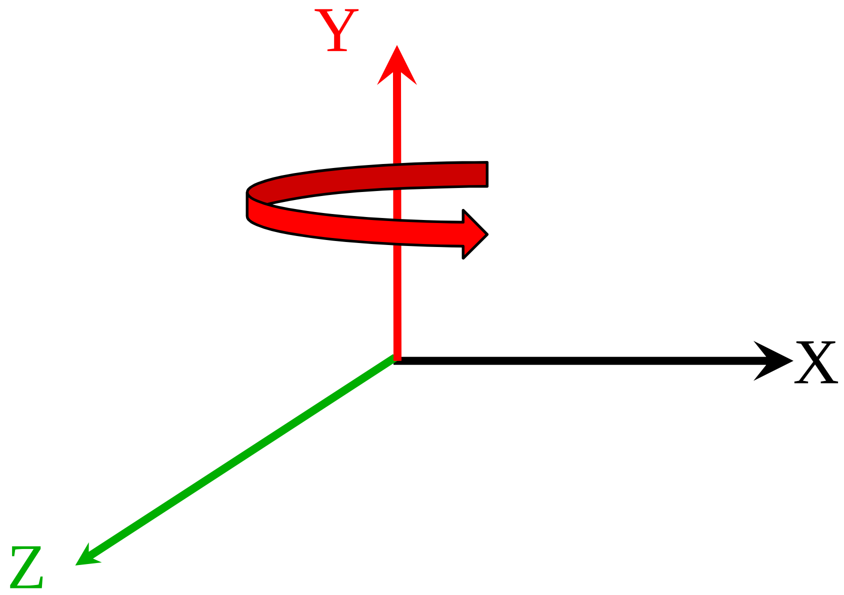
- A set of three gimbals mounted together to allow three degrees of freedom: roll, pitch and yaw.
- When two gimbals rotate around the same axis, the system loses one degree of freedom.

Gimbal Lock



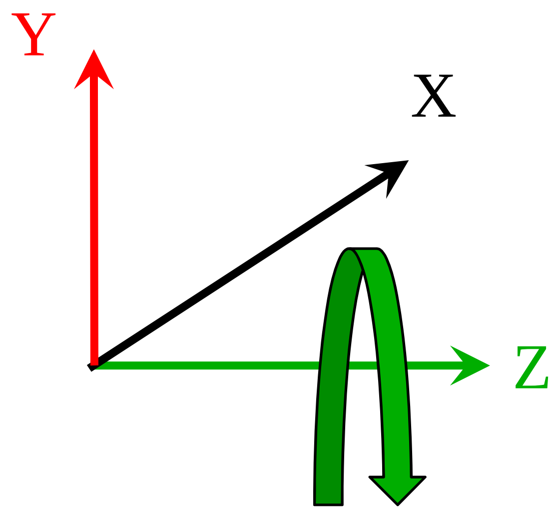
$$\omega = 90°$$

Gimbal Lock



$$\phi = 90°$$

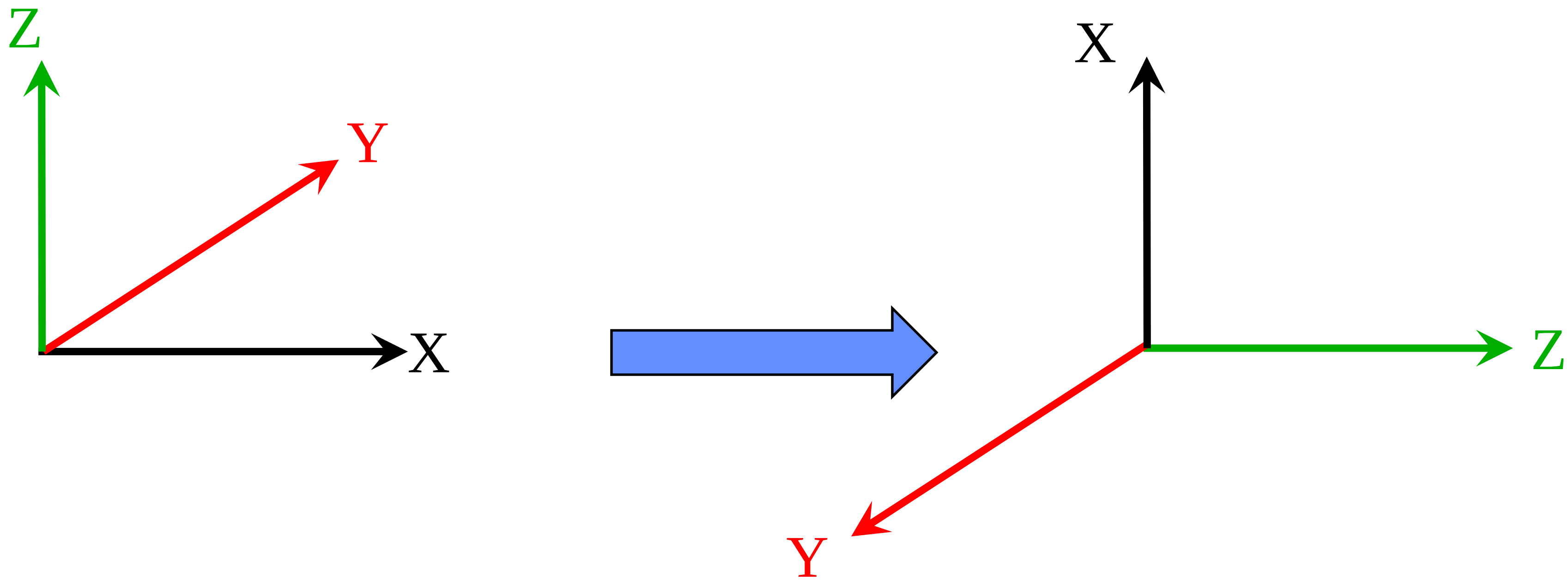
Gimbal Lock



$$\kappa = 90°$$

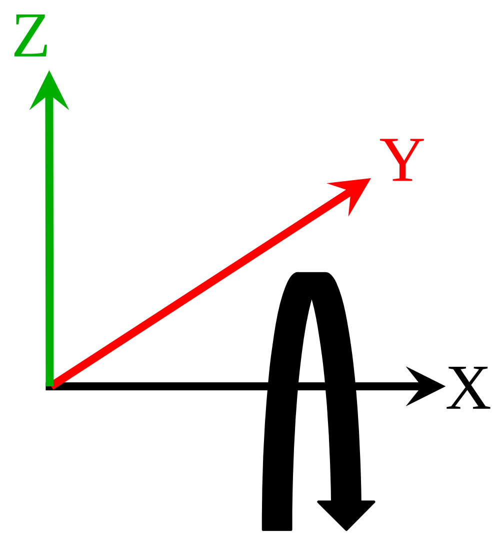
$\omega\ \&\ \kappa$ *rotation angles are around the same axis in space*.

Gimbal Lock



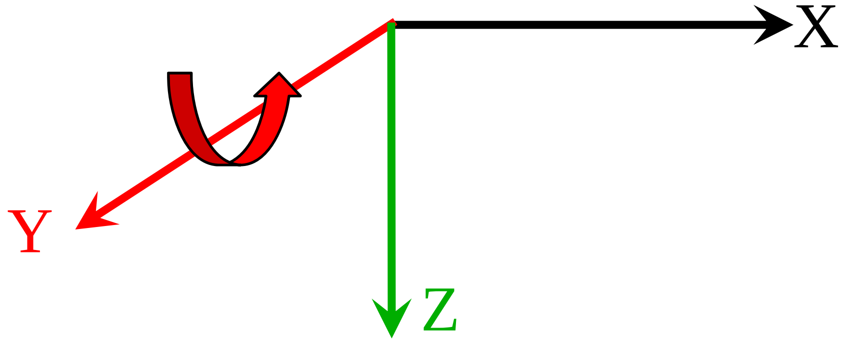
(90°, 90°, 90°)

Gimbal Lock



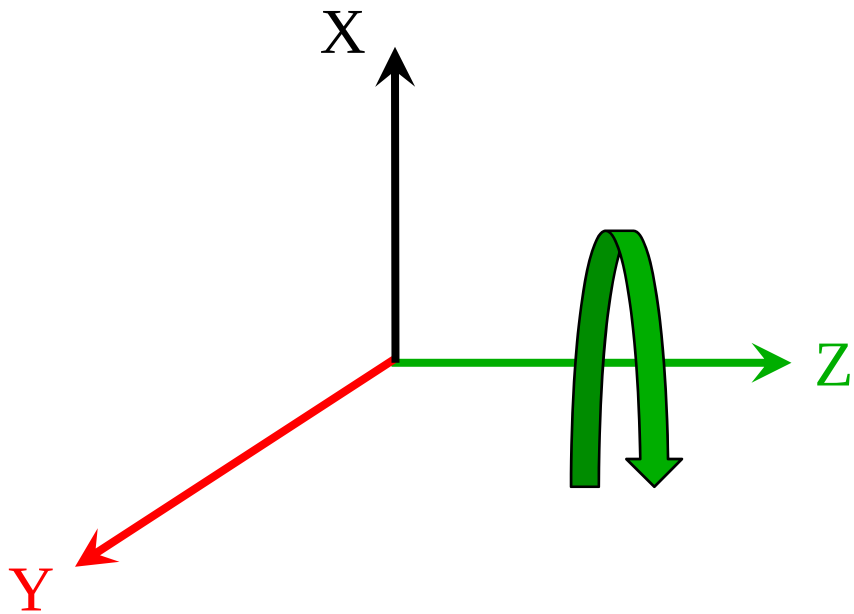
$$\omega = 180°$$

Gimbal Lock



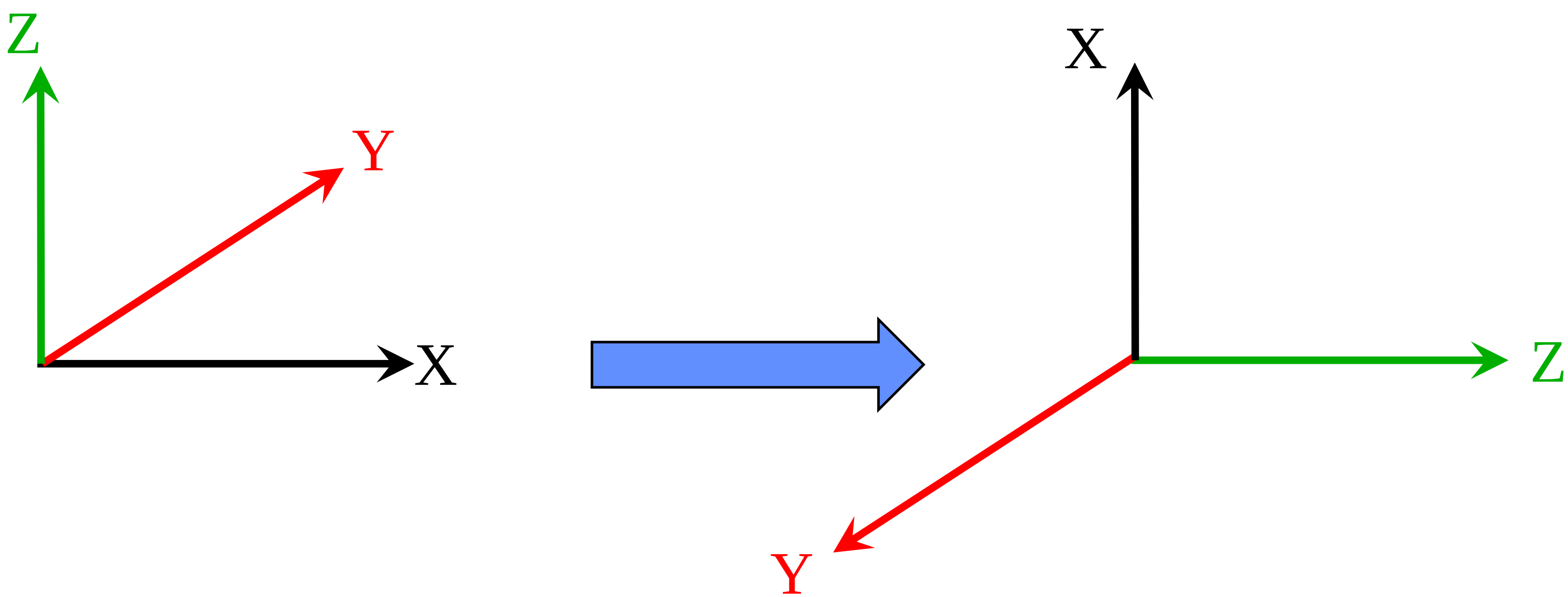
$\phi = 90°$

Gimbal Lock



$\kappa = 0°$

Gimbal Lock



$(90°, 90°, 90°)$ & $(180°, 90°, 0°)$ *are equivalent*!!!

Singularity in the derivation of the rotation angles