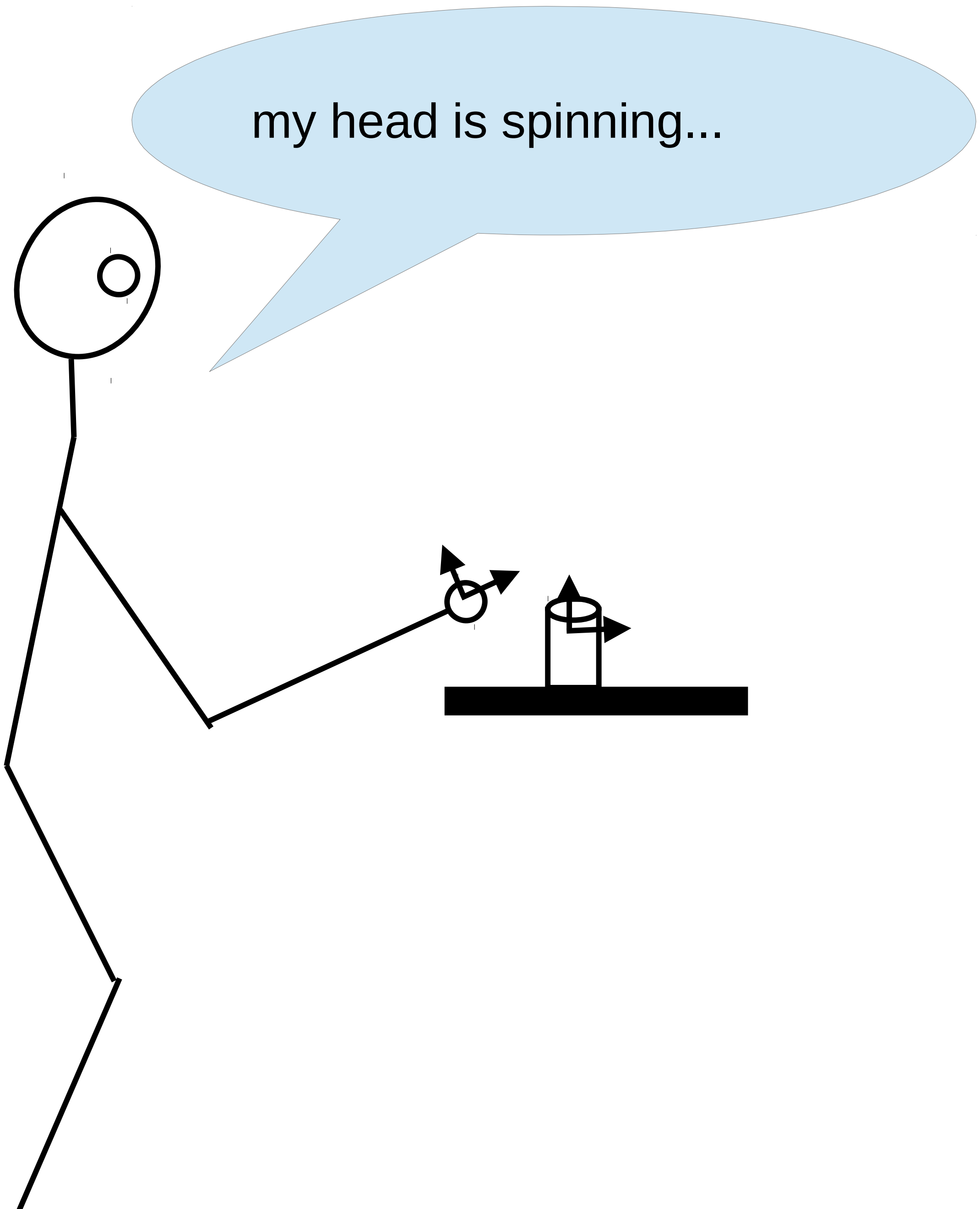
Four different ways to represent rotation



The space of rotations

$$SO(3) = \{R \in R^{3\times 3} \mid RR^T = I, \det(R) = +1\}$$

Special orthogonal group(3):

Why $\det(R) = \pm 1$?

Rotations preserve distance: $\|Rp_1 - Rp_2\| = \|p_1 - p_2\|$

Rotations preserve orientation: $(Rp_1) \times (Rp_2) = R(p_1 \times p_2)$

The space of rotations

$$SO(3) = \{ R \in R^{3\times 3} \mid RR^T = I, \det(R) = +1 \}$$

↑

Special orthogonal group(3):

Why it's a group:

- Closed under multiplication: if $R_1, R_2 \in SO(3)$ then $R_1 R_2 \in SO(3)$
- Has an identity: $\exists I \in SO(3) \text{ s.t. } IR_1 = R_1$
- Has a unique inverse…
- Is associative…

Why orthogonal:

- vectors in matrix are orthogonal

Why it's special: $\det(R) = +1$, NOT $\det(R) = \pm 1$

↑

Right hand coordinate system

Possible rotation representations

You need at least three numbers to represent an arbitrary rotation in SO(3) (Euler theorem). Some three-number representations:

- ZYZ Euler angles
- ZYX Euler angles (roll, pitch, yaw)
- Axis angle

One four-number representation:

- quaternions

ZYZ Euler Angles

$$r_{zyz} = \begin{pmatrix} \phi \\ \theta \\ \psi \end{pmatrix}$$

To get from *A* to *B:*

1. Rotate ϕ about *z* axis

2. Then rotate θ about y axis

3. Then rotate ψ about z axis

$$R_z(\phi) = \begin{pmatrix} \cos\phi & -\sin\phi & 0 \\ \sin\phi & \cos\phi & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

$$R_y(\theta) = \begin{pmatrix} \cos\theta & 0 & \sin\theta \\ 0 & 1 & 0 \\ -\sin\theta & 0 & \cos\theta \end{pmatrix}$$

$$R_z(\psi) = \begin{pmatrix} \cos\psi & -\sin\psi & 0 \\ \sin\psi & \cos\psi & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

ZYZ Euler Angles

Remember that $R_z(\phi)\ \ R_y(\theta)\ \ R_z(\psi)$ encode the desired rotation in the pre-rotation reference frame:

$$R_z(\phi)={}^{pre-rotation}R_{post-rotation}$$

Therefore, the sequence of rotations is concatentated as follows:

$$R_{zyz}(\phi,\theta,\psi) = R_z(\phi)R_y(\theta)R_z(\psi)$$

$$R_{zyz}(\phi,\theta,\psi)=\begin{pmatrix}\cos\phi & -\sin\phi & 0\\ \sin\phi & \cos\phi & 0\\ 0 & 0 & 1\end{pmatrix}\begin{pmatrix}\cos\theta & 0 & \sin\theta\\ 0 & 1 & 0\\ -\sin\theta & 0 & \cos\theta\end{pmatrix}\begin{pmatrix}\cos\psi & -\sin\psi & 0\\ \sin\psi & \cos\psi & 0\\ 0 & 0 & 1\end{pmatrix}$$

$$R_{zyz}(\phi,\theta,\psi)=\begin{pmatrix}c_\phi c_\theta c_\psi - s_\phi s_\psi & -c_\phi c_\theta s_\psi - s_\phi c_\psi & c_\phi s_\theta\\ s_\phi c_\theta c_\psi + c_\phi s_\psi & -s_\phi c_\theta s_\psi + c_\phi c_\psi & s_\phi s_\theta\\ -s_\theta c_\psi & s_\theta s_\psi & c_\theta\end{pmatrix}$$

ZYX Euler Angles (roll, pitch, yaw)

To get from *A* to *B:*

1. Rotate ϕ about *z* axis

2. Then rotate θ about y axis

3. Then rotate ψ about x axis

$$R_z(\phi) = \begin{pmatrix} \cos\phi & -\sin\phi & 0 \\ \sin\phi & \cos\phi & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

$$R_y(\theta) = \begin{pmatrix} \cos\theta & 0 & \sin\theta \\ 0 & 1 & 0 \\ -\sin\theta & 0 & \cos\theta \end{pmatrix}$$

$$R_x(\psi) = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos\psi & -\sin\psi \\ 0 & \sin\psi & \cos\psi \end{pmatrix}$$

$$R_{zyx}\left(\phi,\theta,\psi\right) = R_z(\phi)R_y(\theta)R_x(\psi)$$

$$R_{zyz}\left(\phi,\theta,\psi\right) = \begin{pmatrix} \cos\phi & -\sin\phi & 0 \\ \sin\phi & \cos\phi & 0 \\ 0 & 0 & 1 \end{pmatrix}\begin{pmatrix} \cos\theta & 0 & \sin\theta \\ 0 & 1 & 0 \\ -\sin\theta & 0 & \cos\theta \end{pmatrix}\begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos\psi & -\sin\psi \\ 0 & \sin\psi & \cos\psi \end{pmatrix}$$

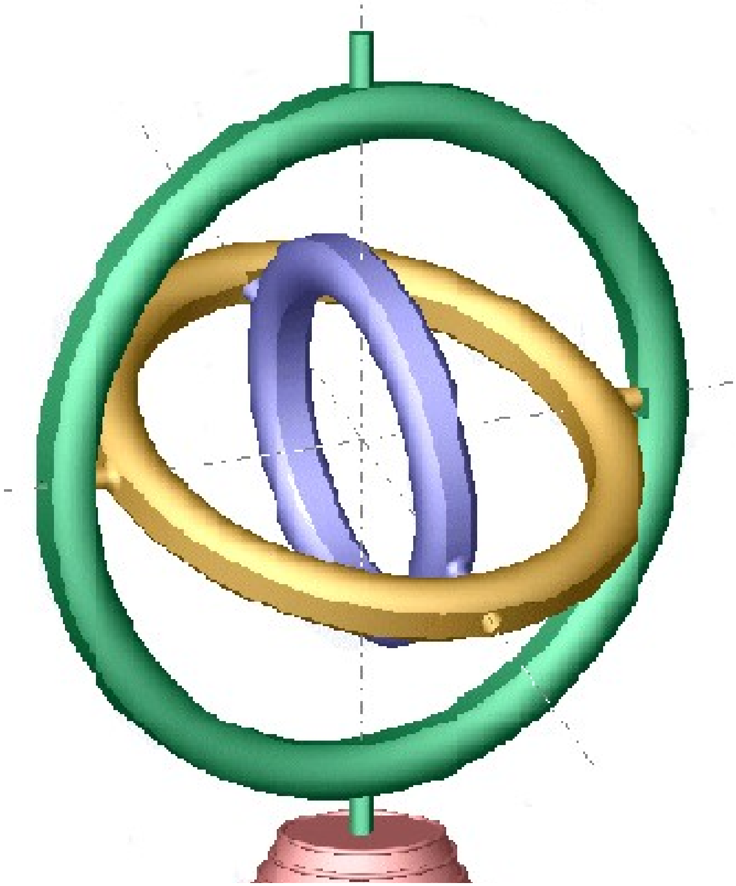
Problems w/ Euler Angles

If two axes are aligned, then there is a “don’t care” manifold of Euler angles that represent the same orientation

- The system loses one DOF

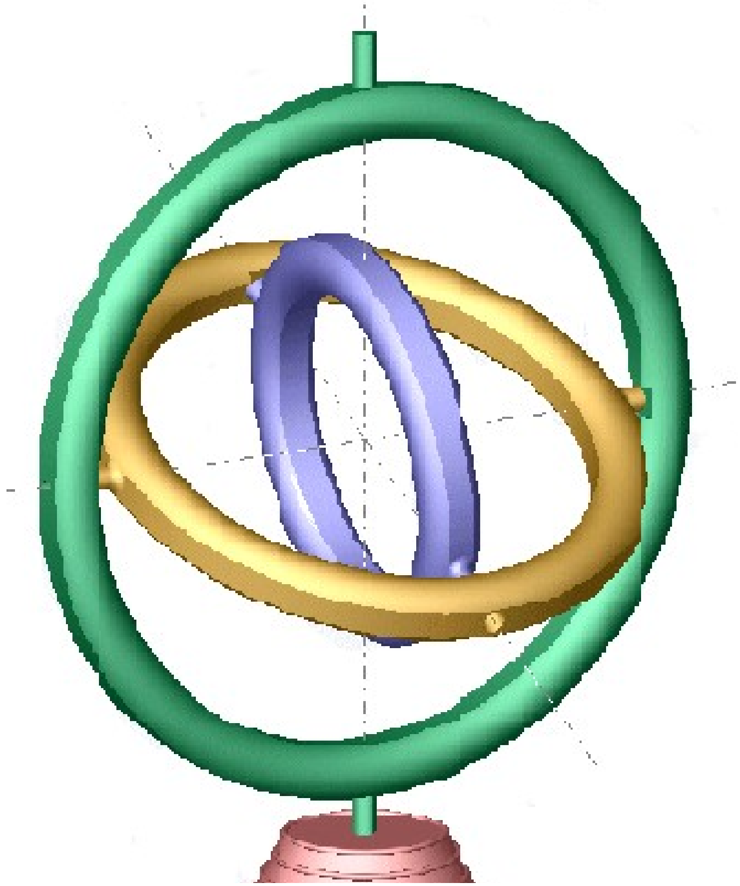
$$r_1 = \begin{pmatrix} 0 \\ 90^\circ \\ 0 \end{pmatrix} \qquad r_2 = \begin{pmatrix} 90^\circ \\ 89^\circ \\ 90^\circ \end{pmatrix}$$

$$r_1 - r_2 = \begin{pmatrix} -90^\circ \\ 1^\circ \\ -90^\circ \end{pmatrix}$$, but the actual distance is 1°



Problem w/ Euler Angles: gimbal lock

1. When a small change in orientation is associated with a large change in rotation representation
2. Happens in “singular configurations” of the rotational representation (similar to singular configurations of a manipulator)
3. This is a problem w/ any Euler angle representation



Problem w/ Euler Angles: gimbal lock

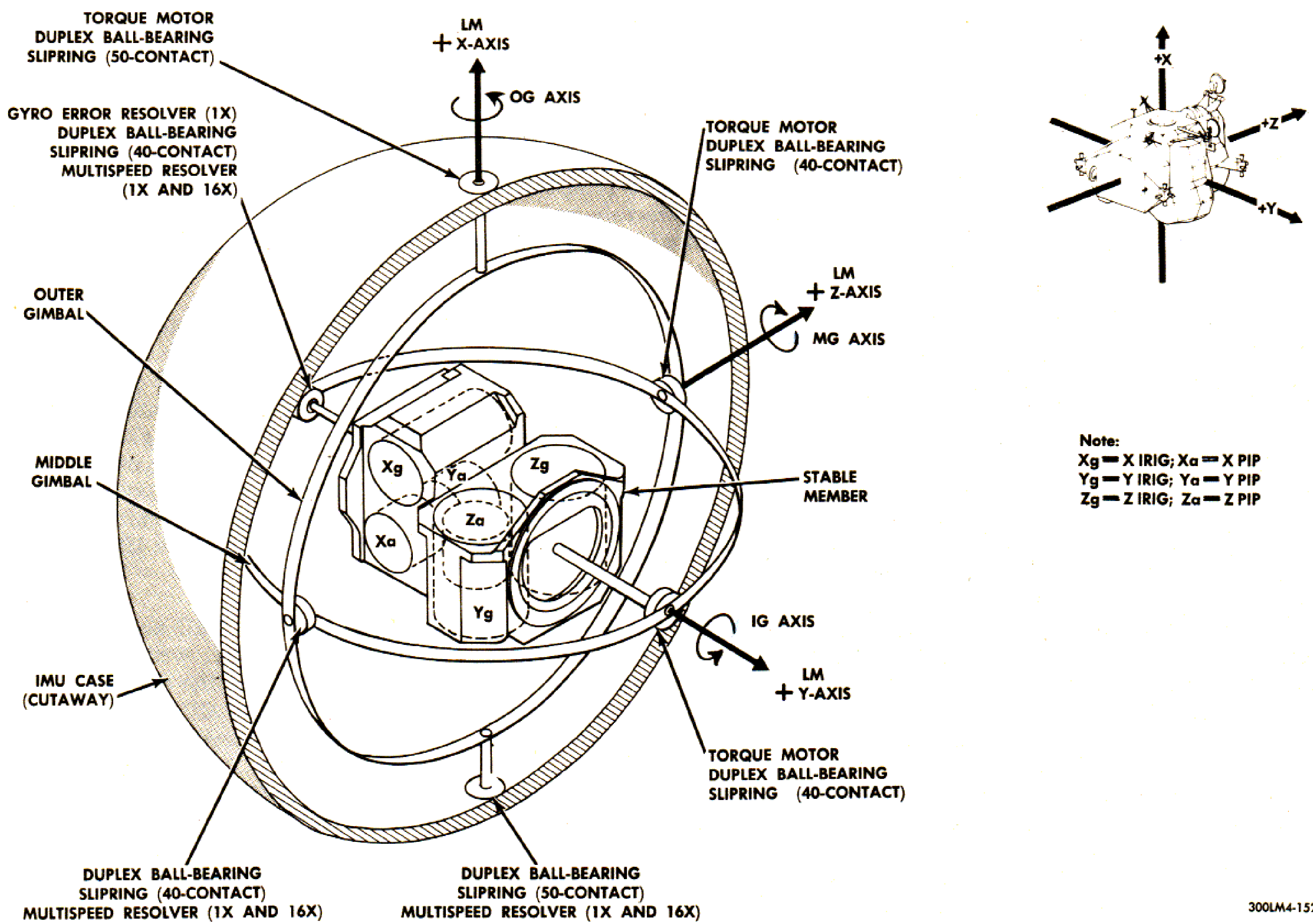
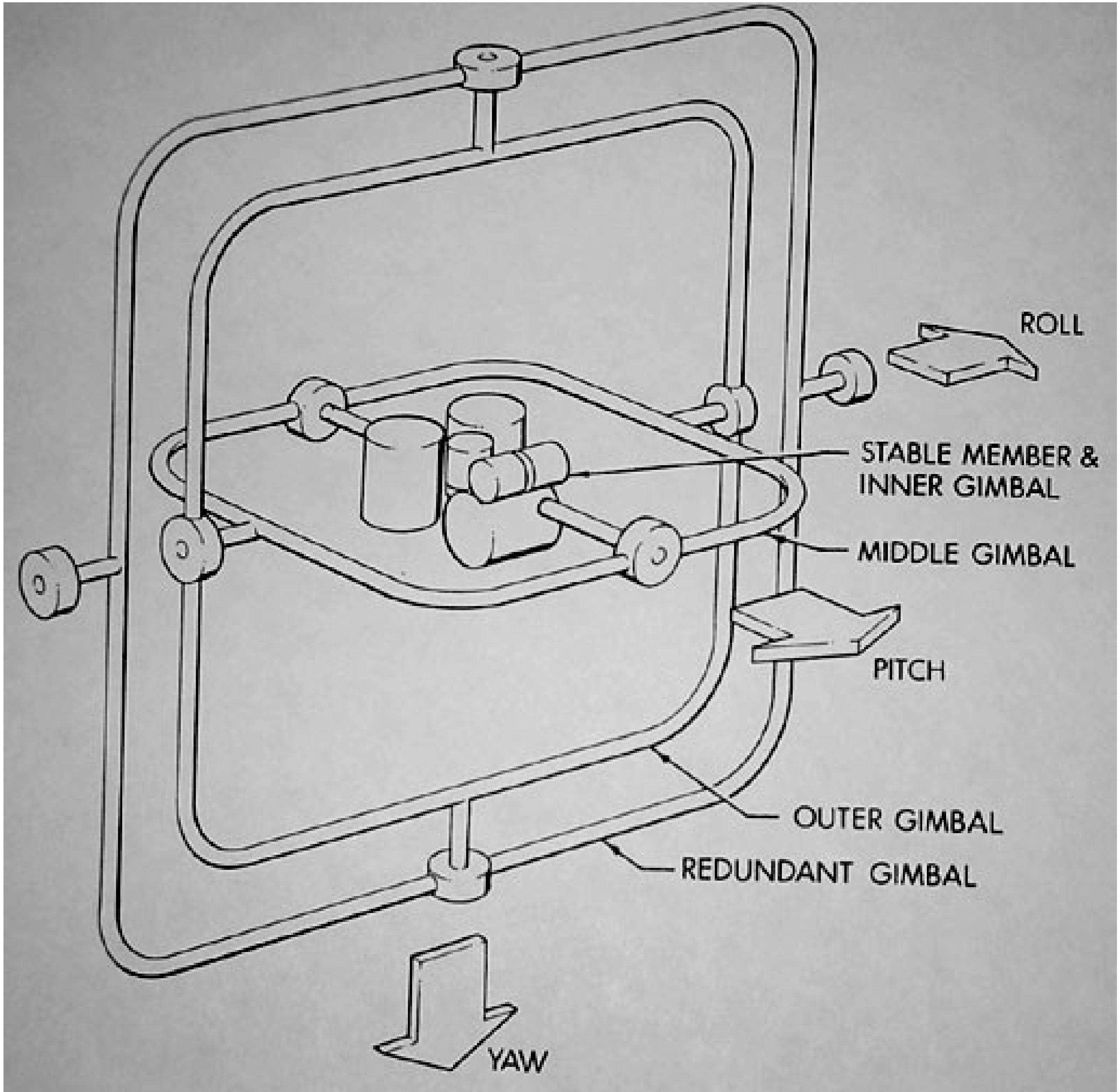


Figure 2.1-24. IMU Gimbal Assembly

Problem w/ Euler Angles: gimbal lock



Axis-angle representation

Theorem: (Euler). Any orientation, $R \in SO(3)$, is equivalent to a rotation about a fixed axis, $\omega \in R^3$, through an angle $\theta \in [0,2\pi)$

(also called *exponential coordinates*)

Axis: $\mathbf{k} = \begin{pmatrix} k_x \\ k_y \\ k_z \end{pmatrix}$ Angle: θ

Converting to a rotation matrix:

$$R_{\mathbf{k}\theta} = e^{S(\mathbf{k})\theta} = I + S(\mathbf{k})\sin(\theta) + S(\mathbf{k})^2(1 - \cos(\theta))$$

$$= [\text{that equation in the book...}]$$

Rodrigues' formula

Axis-angle representation

Converting to axis angle:

Magnitude of rotation:

$$\theta = |k| = \cos^{-1}\left(\frac{trace(R)-1}{2}\right)$$

Axis of rotation:

$$\hat{k} = \frac{1}{2\sin\theta}\begin{pmatrix} r_{32} - r_{23} \\ r_{13} - r_{31} \\ r_{21} - r_{12} \end{pmatrix}$$

Where:

$$R = \begin{pmatrix} r_{11} & r_{12} & r_{13} \\ r_{12} & r_{22} & r_{23} \\ r_{13} & r_{23} & r_{33} \end{pmatrix}$$

and:

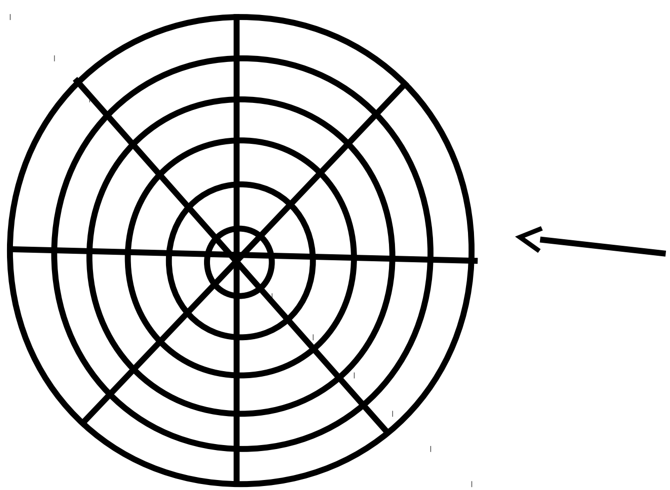
$$trace(R) = r_{11} + r_{22} + r_{33}$$

Axis-angle problems

Still suffers from the "edge" and distance preserving problems of Euler angles:

$$r_1 = \begin{pmatrix} 0 \\ 0 \\ 179^\circ \end{pmatrix} \qquad r_2 = \begin{pmatrix} 0 \\ 0 \\ -179^\circ \end{pmatrix} \qquad r_1 - r_2 = \begin{pmatrix} 0 \\ 0 \\ 358^\circ \end{pmatrix}$$

, but the actual distance is 2°



Distance metric changes as you get further from origin.

Axis-angle representation

Axis angle is can be encoded by just three numbers instead of four:

If $k \neq 0$ then $\hat{k} = \frac{k}{|k|}$ and $\theta = |k|$

If the three-number version of axis angle is used, then

$$R_0 = I$$

For most orientations, R_k , is unique.

For rotations of 180°, there are two equivalent representations:

If $|k| = 180^\circ$ then $R_k = R_{-k}$

Projection distortions

Example: differencing rotations

Calculate the difference between these two rotations:

$$k_1 = \begin{pmatrix} \pi/2 \\ 0 \\ 0 \end{pmatrix} \qquad k_2 = \begin{pmatrix} 0 \\ \pi/2 \\ 0 \end{pmatrix}$$

This is NOT the right answer:

$$k_1 - k_2 = \begin{pmatrix} \pi/2 \\ -\pi/2 \\ 0 \end{pmatrix}$$

According to that, this is the magnitude of the difference:

$$\left| k_1 - k_2 \right| = \frac{\pi}{\sqrt{2}} = 127.27^\circ$$

Example: differencing rotations

Convert to rotation matrices to solve this problem:

$$k_1 = \begin{pmatrix} \pi/2 \\ 0 \\ 0 \end{pmatrix} \qquad k_2 = \begin{pmatrix} 0 \\ \pi/2 \\ 0 \end{pmatrix}$$

$${}^1R_2 = {}^BR_1^T \, {}^BR_2$$

$${}^bR_1 = R_x(\pi/2) = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos(\pi/2) & -\sin(\pi/2) \\ 0 & \sin(\pi/2) & \cos(\pi/2) \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & -1 \\ 0 & 1 & 0 \end{pmatrix}$$

$${}^bR_2 = R_y(\pi/2) = \begin{pmatrix} \cos(\pi/2) & 0 & \sin(\pi/2) \\ 0 & 1 & 0 \\ -\sin(\pi/2) & 0 & \cos(\pi/2) \end{pmatrix} = \begin{pmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ -1 & 0 & 0 \end{pmatrix}$$

$${}^1R_2 = {}^BR_1^T \, {}^BR_2 = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & -1 & 0 \end{pmatrix} \begin{pmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ -1 & 0 & 0 \end{pmatrix} = \begin{pmatrix} 0 & 0 & 1 \\ -1 & 0 & 0 \\ 0 & -1 & 0 \end{pmatrix}$$

$$\theta = \cos^{-1}\left(\frac{trace(R)-1}{2}\right) = \cos^{-1}\left(-\frac{1}{2}\right) = \tfrac{2}{3}\pi \qquad \hat{k} = \frac{1}{2\sin\theta}\begin{pmatrix} r_{32}-r_{23} \\ r_{13}-r_{31} \\ r_{21}-r_{12} \end{pmatrix} = \frac{1}{\sqrt{3}}\begin{pmatrix} -1 \\ 1 \\ -1 \end{pmatrix} \qquad k = \frac{\frac{2}{3}\pi}{\sqrt{3}}\begin{pmatrix} -1 \\ 1 \\ -1 \end{pmatrix}$$

Quaternions

So far, rotation matrices seem to be the most reliable method of manipulating rotations. But there are problems:

- Over a long series of computations, numerical errors can cause these 3x3 matrices to no longer be orthogonal (you need to “orthogonalize” them from time to time).
- Although you can accurately calculate rotation differences, you can’t interpolate over a difference.’
    - Suppose you wanted to smoothly rotate from one orientation to another – how would you do it?

Answer: quaternions…

Quaternions

Generalization of complex numbers: $Q = q_0 + iq_1 + jq_2 + kq_3$

$$Q = (q_0, q)$$

Essentially a 4-dimensional quantity

Properties of complex dimensions:

$$ii = jj = kk = ijk = -1 \qquad jk = -kj = i$$

$$ij = -ji = k \qquad ki = -ik = j$$

Multiplication:

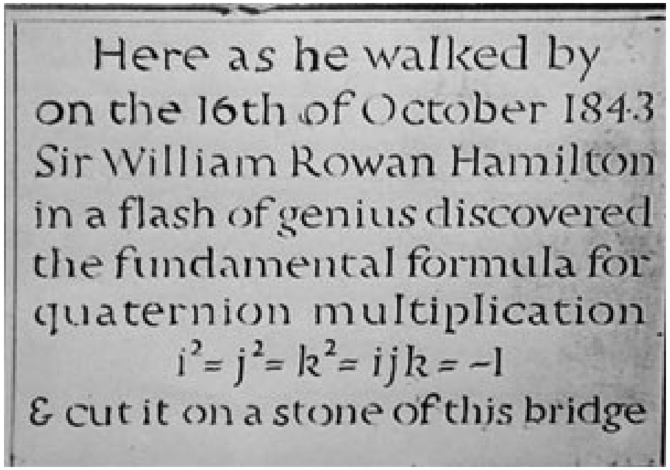
$$QP = (q_0 + iq_1 + jq_2 + kq_3)(p_0 + ip_1 + jp_2 + kp_3)$$

$$QP = (p_0 q_0 - p \cdot q,\ p_0 q + q_0 p + p \times q)$$

Complex conjugate: $Q^* = (q_0, q)^* = (q_0, -q)$

Quaternions

Invented by Hamilton in 1843:



Along the royal canal in Dublin…

Quaternions

Let's consider the set of unit quaternions: $Q^2 = q_0{}^2 + q_1{}^2 + q_2{}^2 + q_3{}^2 = 1$

This is a four-dimensional hypersphere, *i.e.* the 3-sphere S^3

The identity quaternion is: $Q = (1,0)$

Since: $QQ^* = (q_0, q)(q_0, -q) = (q_0 q_0 - q^2, q_0 q - q_0 q + q \times q) = (1,0)$

Therefore, the inverse of a unit quaternion is: $Q^* = Q^{-1}$

Quaternions

Associate a rotation with a unit quaternion as follows:

Given a unit axis, $\hat{k}$, and an angle, θ: ⟵ (just like axis angle)

The associated quaternion is: $$Q_{\hat{k},\theta} = \left(\cos\left(\frac{\theta}{2}\right), \hat{k}\sin\left(\frac{\theta}{2}\right) \right)$$

Therefore, Q represents the same rotation as $-Q$

Let ${}^{i}P = \left(0, {}^{i}p\right)$ be the quaternion associated with the vector ${}^{i}p$

You can rotate ${}^{a}P$ from frame *a* to *b*: ${}^{b}P = Q_{ba}\, {}^{a}P Q_{ba}{}^{*}$

Composition: $Q_{ca} = Q_{cb} Q_{ba}$

Inversion: $Q_{cb} = Q_{ca} Q_{ba}{}^{-1}$

Example: Quaternions

Rotate $\quad {}^{a}P = \left(0, \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix} \right)$ by $\quad Q = \left(\frac{1}{\sqrt{2}}, \begin{pmatrix} 0 \\ \frac{1}{\sqrt{2}} \\ 0 \end{pmatrix} \right)$

$$
{}^{b}P = Q\,{}^{a}P\,Q^{*} = \left(\frac{1}{\sqrt{2}}, \begin{pmatrix} 0 \\ \frac{1}{\sqrt{2}} \\ 0 \end{pmatrix} \right) \left(0, \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix} \right) \left(\frac{1}{\sqrt{2}}, \begin{pmatrix} 0 \\ -\frac{1}{\sqrt{2}} \\ 0 \end{pmatrix} \right)
$$

$$
= \left(\frac{1}{\sqrt{2}}, \begin{pmatrix} 0 \\ \frac{1}{\sqrt{2}} \\ 0 \end{pmatrix} \right) \left(0, \begin{pmatrix} \frac{1}{\sqrt{2}} \\ 0 \\ -\frac{1}{\sqrt{2}} \end{pmatrix} \right)
$$

$$
= \left(0, \begin{pmatrix} \frac{1}{2} \\ 0 \\ -\frac{1}{2} \end{pmatrix} + \begin{pmatrix} -\frac{1}{2} \\ 0 \\ -\frac{1}{2} \end{pmatrix} \right) = \left(0, \begin{pmatrix} 0 \\ 0 \\ -1 \end{pmatrix} \right)
$$

Example: Quaternions

Find the difference between these two axis angle rotations:

$$k_1 = \begin{pmatrix} \pi/2 \\ 0 \\ 0 \end{pmatrix} \qquad k_2 = \begin{pmatrix} 0 \\ \pi/2 \\ 0 \end{pmatrix}$$

$$\sin\left(\frac{\pi}{4}\right) = \cos\left(\frac{\pi}{4}\right) = \frac{1}{\sqrt{2}} \qquad Q_{cb} = \left(\frac{1}{\sqrt{2}}, \begin{pmatrix} 0 \\ \frac{1}{\sqrt{2}} \\ 0 \end{pmatrix}\right) \quad Q_{ba} = \left(\frac{1}{\sqrt{2}}, \begin{pmatrix} \frac{1}{\sqrt{2}} \\ 0 \\ 0 \end{pmatrix}\right)$$

$$QP = \left(p_0 q_0 - p \cdot q,\ p_0 q + q_0 p + p \times q\right)$$

$$Q_{cb} = Q_{ca} Q_{ba}^{-1} = \left(\frac{1}{\sqrt{2}}, \begin{pmatrix} 0 \\ \frac{1}{\sqrt{2}} \\ 0 \end{pmatrix}\right)\left(\frac{1}{\sqrt{2}}, \begin{pmatrix} -\frac{1}{\sqrt{2}} \\ 0 \\ 0 \end{pmatrix}\right)$$

$$= \left(\frac{1}{2}, \frac{1}{\sqrt{2}}\begin{pmatrix} -\frac{1}{\sqrt{2}} \\ \frac{1}{\sqrt{2}} \\ -\frac{1}{\sqrt{2}} \end{pmatrix}\right) = \left(\frac{1}{2}, \begin{pmatrix} -\frac{1}{2} \\ \frac{1}{2} \\ -\frac{1}{2} \end{pmatrix}\right)$$

$$\theta_{cb} = \cos^{-1}\left(\frac{1}{2}\right) = \frac{2}{3}\pi$$

$$k_{cb} = \begin{pmatrix} -\frac{1}{\sqrt{3}} \\ \frac{1}{\sqrt{3}} \\ -\frac{1}{\sqrt{3}} \end{pmatrix}$$

Quaternions: Interpolation

Suppose you're given two rotations, R_1 and R_2

How do you calculate intermediate rotations?

$$R_i = \alpha R_1 + (1-\alpha) R_2$$

← This does not even result in a rotation matrix

Do quaternions help?

$$Q_i = \frac{\alpha Q_1 + (1-\alpha) Q_2}{\left|\alpha Q_1 + (1-\alpha) Q_2\right|}$$

Suprisingly, this actually works

- Finds a geodesic

This method normalizes automatically (SLERP):

$$Q_i = \frac{Q_1 \sin(1-\alpha)\Omega + Q_2 \sin \alpha\Omega}{\sin \Omega}$$

Supplementary

ZYX Euler Angles (roll, pitch, yaw)

In Euler angles, the each rotation is imagined to be represented in the post-rotation coordinate frame of the last rotation

In Fixed angles, all rotations are imagined to be represented in the original (fixed) coordinate frame.

ZYX Euler angles can be thought of as:

1. ZYX Euler
2. XYZ Fixed

$$R_{zyx}\left(\phi,\theta,\psi\right) = R_z(\phi)R_y(\theta)R_x(\psi)$$