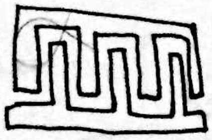
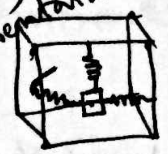
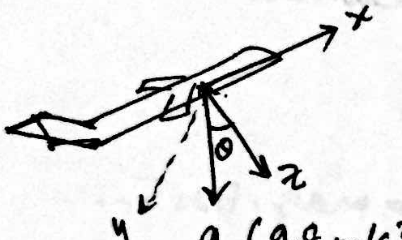
IMU Basics

IMU

acc. (3D)	gyro (3D)	compass (3D)
↓	↓	↓
$\left.\frac{d^2 l}{dt^2}\right\|_{x,y,z}$ l = loc.	$\left.\frac{d\theta}{dt}\right\|_{x,y,z}$ θ = orientation	$\propto$ w.r.t. ~~global~~ magnetic north $\|_{x,y,z}$

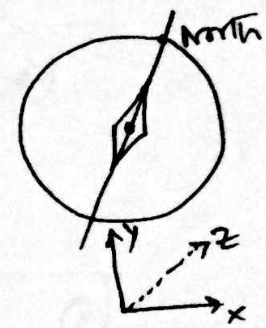






g (9.8 m/s²)

⊖ Orientation of an object at all times will give me rotation.



$a_z = \vec{g}\cos\theta$

$a_x = \vec{g}\sin\theta$

$a_y = 0.$

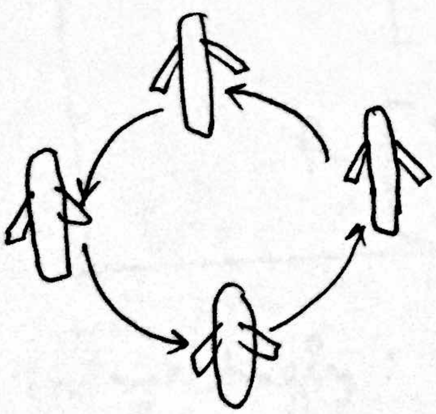
⊖ Orientation = 3D misalighment between local & global coord. frames.

Mag = $\langle M_x, M_n, M_z \rangle$.

$\boxed{\text{global}}$

$$\begin{bmatrix} x_l \\ y_l \\ z_l \end{bmatrix} = \begin{bmatrix} \text{Rot} \\ \text{Matrix} \end{bmatrix} \begin{bmatrix} x_g \\ y_g \\ z_g \end{bmatrix}$$

⊖



Linear acc (centripetel force) will be measured by acc.

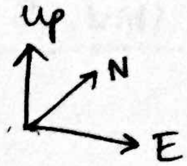
$\boxed{\text{local.}}$

⊖ gyro gives

$$\begin{bmatrix} x_t \\ y_t \\ z_t \end{bmatrix} = \begin{bmatrix} \text{gyro} \\ \text{3D} \\ \text{matrix} \end{bmatrix} \begin{bmatrix} x_{t-1} \\ y_{t-1} \\ z_{t-1} \end{bmatrix}$$

$\boxed{\text{local}}$

→ How to compute 3D orientation?



① Fix global coordinate frame.

~~② Look at acc. to determine how much of rotation needed~~

② To compute misalignment (3D orientation), we need some measurable quantity which is known or fixed in the global reference frame, and measurable in the local reference frame

⇓

gravity, ~~magneton~~ mag. North.

③ Rotate object until local measure of gravity aligns with $-Z_{local}$ axis, i.e., $a_z = -9.8\ m/s^2$

④ Rotate object until ~~magneton~~ compass is 0° N.
i.e., $M_y = 0^\circ$.

⑤ Create a rotation matrix

$$R = R_{mag} \cdot R_{gravity}$$

⑥ 3D orientation $= R^{-1}$.

$$\begin{bmatrix} \text{3R} \\ \text{Rot} \\ \text{Matrix} \\ R_g \end{bmatrix} \begin{bmatrix} a_x \\ a_y \\ a_z \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 9.8 \end{bmatrix}$$

$$\begin{bmatrix} R_m \end{bmatrix} \begin{bmatrix} R_g \end{bmatrix} \begin{bmatrix} a_x \\ a_y \\ a_z \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 9.8 \end{bmatrix}$$

$$R_m \begin{bmatrix} m_x \\ m_y \\ m_z \end{bmatrix} = \begin{bmatrix} 0 \\ |N| \\ 0 \end{bmatrix}$$

$\therefore \quad R = R_m R_g = $ Orientation matrix

$$R . \begin{bmatrix} a_x \\ a_y \\ a_z \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 9.8 \end{bmatrix}$$

$$R \begin{bmatrix} m_x \\ m_y \\ m_z \end{bmatrix} = \begin{bmatrix} 0 \\ N \\ 0 \end{bmatrix}$$

Interestingly: $\begin{bmatrix} l_x \\ l_y \\ l_z \end{bmatrix} = R \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}$

IMU noise Integration

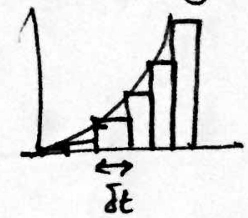
Data samples : $D_1 \quad D_2 \quad D_3 \quad \ldots \quad D_t$

Noisy data : $(D_1 + N_1) \quad (D_2 + N_2) \quad (D_3 + N_3) \quad \ldots \quad (D_t + N_t)$

where N is white noise.

→ To understand the effect of noise on the integral of the data, i.e., $\int_0^t D_t \, dt$. → gyro → rotation; → acc → velocity.

→ so we really need to understand integral of white noise sig:



$$\int_0^t \epsilon(t)\, dt = \delta t \sum_{i=1}^{n} N_i$$

↳ $t = n . \delta t$

→ Now N_i is identically distributed, s.t. $E[N_i] = E[N] = 0$

and $Var(N_i) = Var(N) = \sigma^2$.

By definition, a white noise sequence has

$$Cov(N_i, N_j) = 0 \quad \text{for } i \neq j$$

$$\therefore \quad E\left[\int_0^t \epsilon(t)\, dt\right] = \delta t . n . E[n] = 0$$

$$Var\left[\int_0^t \epsilon(t)\, dt\right] = \delta t^2 . n . \sigma^2 = \delta t . t . \sigma^2$$

$\therefore$ Std. dev. $\sigma_\theta(t) = \sqrt{t . \delta t \, \sigma^2}$

which grows proportionally to $\sqrt{t}$

Angle Random Walk (ARW) $= \sigma_\theta(1)$

which is degrees deviated in $\sqrt{1}$ time unit. Ex. $0.2° / \sqrt{hour}$