Announcements

- **MP4** is out. Due on Nov 6 @ 11:59pm.

Final project upcoming deadlines:

- Nov 13, a short video of your progress.
- Dec 16, final project presentation.

Grades are out for:

- Midterm 1
- Project abstract and picture-title

Tracking Systems in VR

What do we want to track? Think about rigid bodies:

1. Head wearing HMD
2. Eyes
3. Palms of hands
4. Fingers
5. Entire body
6. Movable objects - controller, coffee cup, desk
7. Other people in the space

Tracking Systems in VR

What do we want to track:

For each rigid body, estimate

Rotation:

Position:

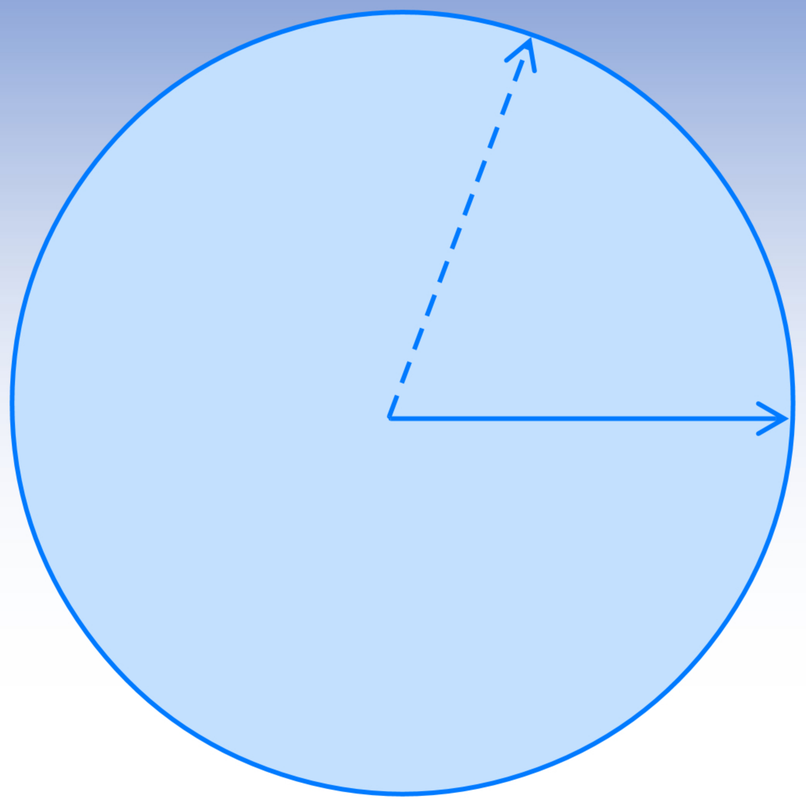
OR

Homogeneous transformation matrix:

Task:

Tracking Systems in VR: Estimating 2D Orientation

easy ?



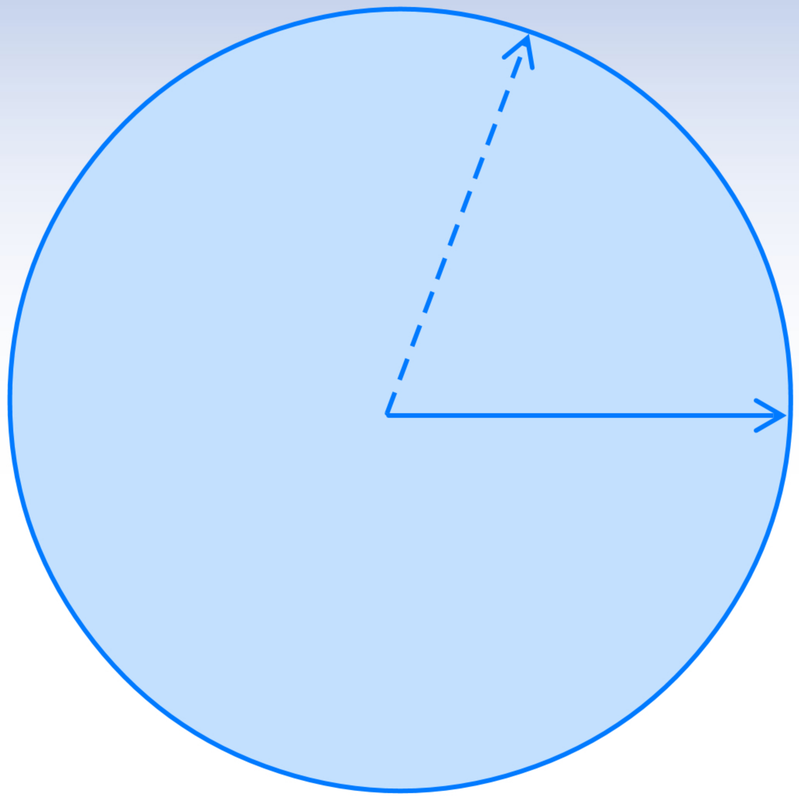
Initial conditions

For constant velocity

Discrete-time approximation

For variable velocity

Tracking Systems in VR: Estimating 2D Orientation



Discrete-time approximation

$$\theta_k \approx \theta_0 + \sum_{i=1}^{k} \Delta\theta_i =$$

$$= \theta_0 + \sum_{i=1}^{k} \omega_i \Delta t$$

$\Delta t = 1$ msec

$\omega_i = \omega((i-1)\Delta t)$

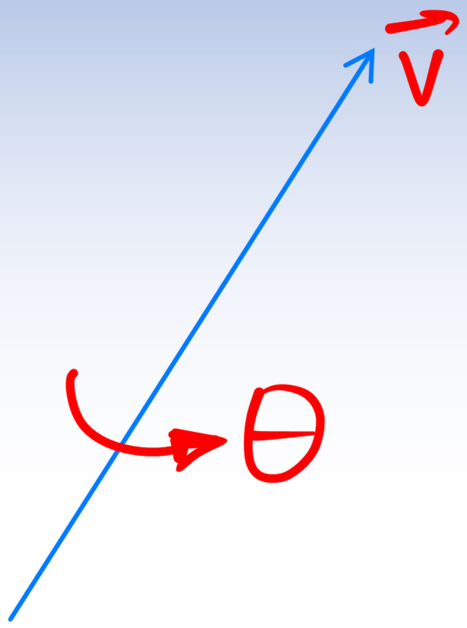
Estimation given sensor readings ω_i every Δt

Q1: does it matter where we put the gyroscope on the disk?

Q2: How many lines of code to implement this estimation?

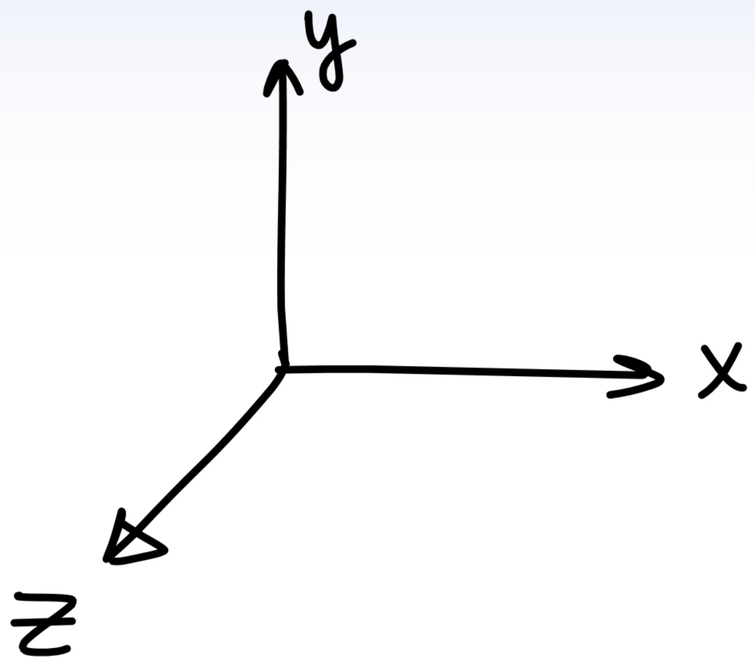
Tracking Systems in VR: Estimating 3D Orientation

Axis-angle:



3-axis gyroscope measures:

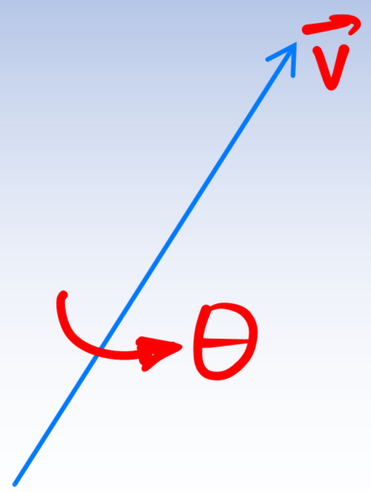
$\omega =$



Examples of VR systems with IMU only tracking:

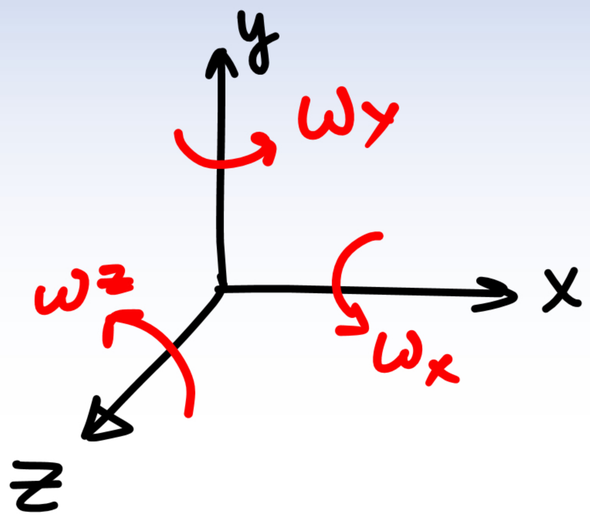
Tracking Systems in VR: Estimating 3D Orientation

Axis-angle: $(\vec{v}, \theta)$



3-axis gyroscope measures:

$\vec{\omega} = (\omega_x, \omega_y, \omega_z)$



How large is ΔQ_i ?

Tracking Systems in VR: Estimating 3D Orientation

Integrate sensor readings to estimate orientation:

Recall 2D: $\hat{\theta}_k = \theta_0 + \sum_{i=1}^{k} \Delta\hat{\theta}_i =$

$= \theta_0 + \Delta\hat{\theta}_1 + \Delta\hat{\theta}_2 + \ldots + \Delta\hat{\theta}_k$

3D: $\hat{Q}_k =$

Recursively/Incrementally: 2D: $\hat{\theta}_{current} =$

3D: $\hat{Q}_{current} =$

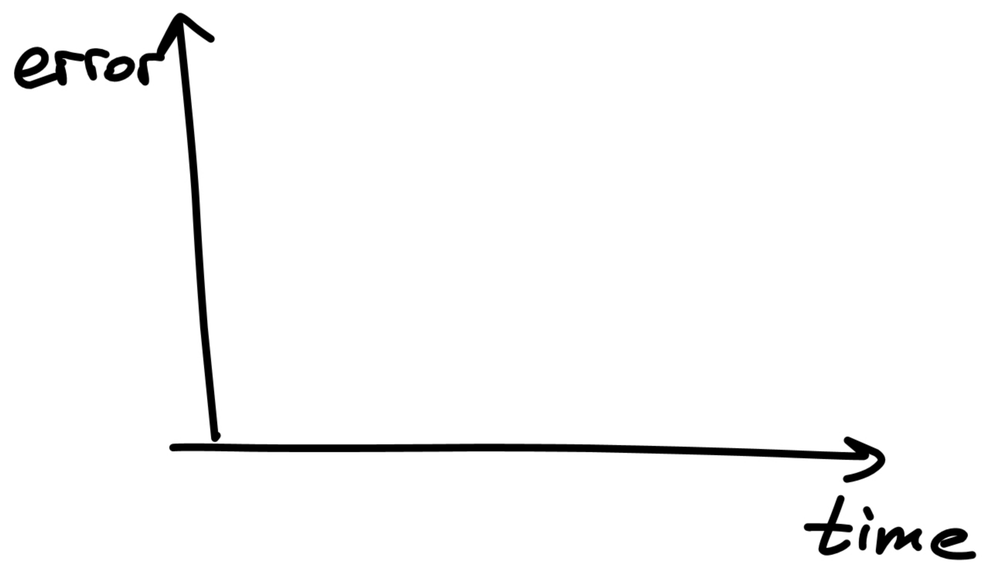
Problem:

Estimating 3D Orientation: Drift Correction

Expression for drift error:

2D: $d_k =$

3D: $d_k =$



An unusual quantity to have for an error!

Estimating 3D Orientation: Drift Correction

Correcting for drift errors challenge:

1) Use another sensor _____

2) Gradually apply corrections

- Fast enough to _____

- Slow enough to _____

Estimating 3D Orientation: Drift Correction

Separate the rotational drift error into **two components:**

1)

To correct:

2)

To correct: