Introduction to Autonomous Electric Vehicles

Lecture 2

- Vehicle Modeling
- Autonomy/ Control architecture
- Introduction to sensors
- Localization

EEE department, BITS-Goa

31-March-2021

Modular EV

Drive-by-wire systems

Electrical rather than mechanical for steering/ throttle/brake

Already present in

- Adaptive Cruise control
- Electronic stability control
- Lane-departure warning

https://m.youtube.com/watch?v=_tWXu372O3k

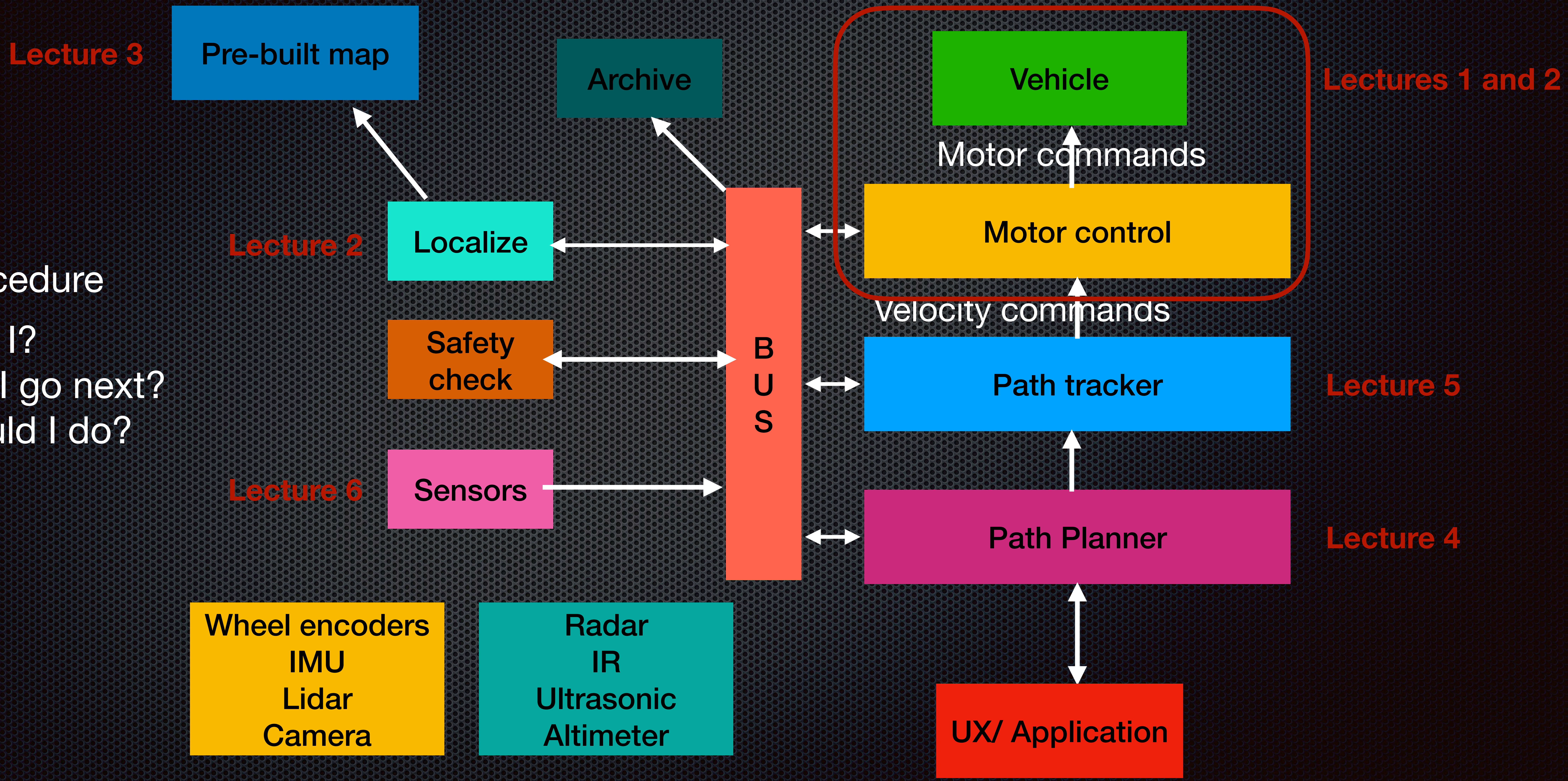
Lecture plan

- Recap of Vehicle Modeling
- Autonomy and control architecture
- Need for Closed-loop feedback
- Introduction to Sensors
  - Incremental wheel encoders
    - Vehicle drift
  - Lidar
    - 2D lidar simulator
  - Others: IMU, Radars, IR, Ultrasonic, Cameras (Lec 6)
- Localization
  - Lidar-based
  - Sensor fusion and Kalman filters

Autonomy and Control: Pub-Sub architecture

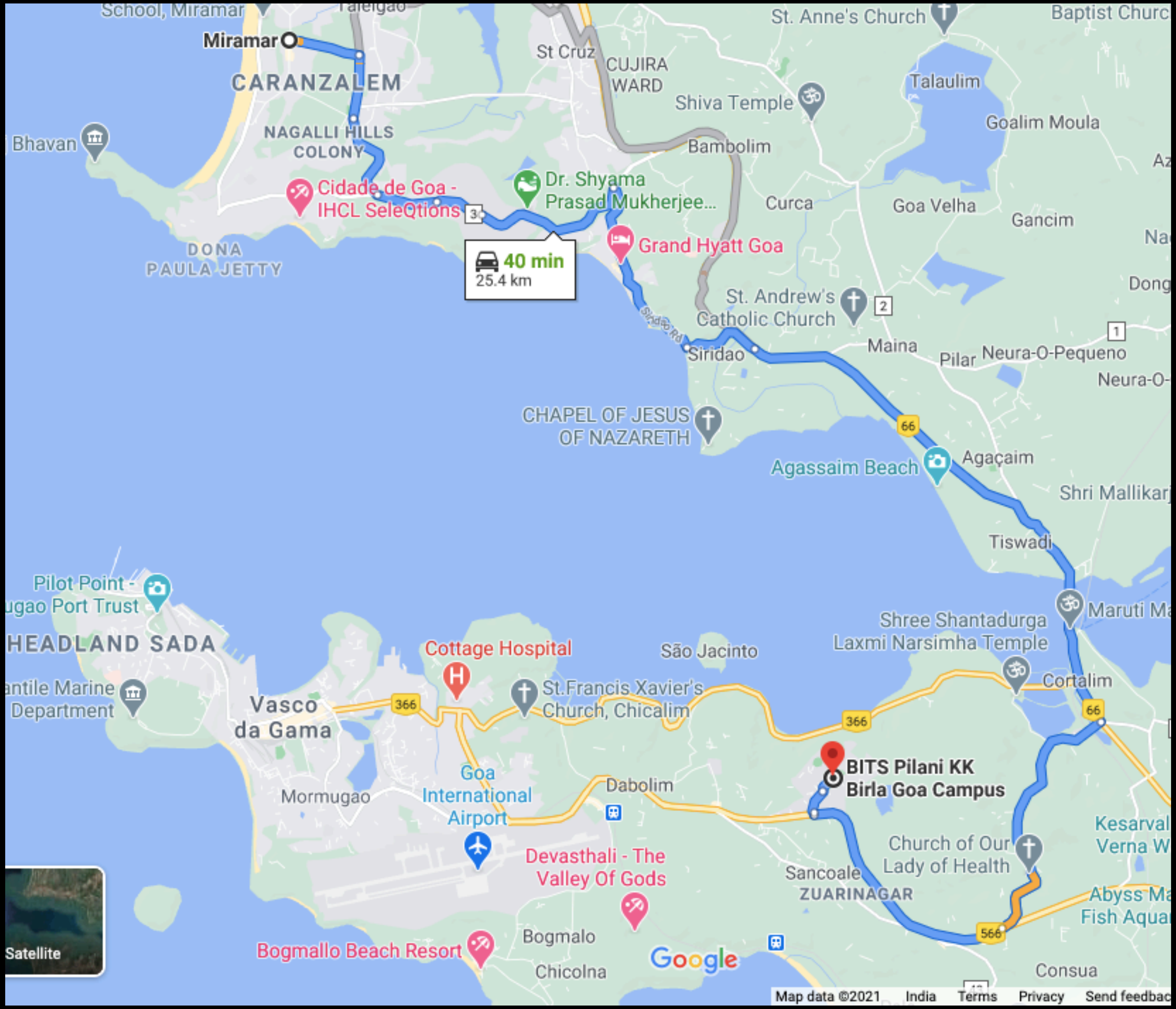
Iterative procedure

1. Where am I?
2. Where do I go next?
3. What should I do?



Global Mission planning

Go from campus to Miramar



High-level estimates

Locally safe planning

Vehicle action in immediate time horizon

- More annotations/ Semantically richer
- Suited for quick Replanning

Unicycle Model

Initially Robot is at A

In reference frame robot pose is (x, y, θ)

At velocity v, Robot moves to B in time dt

Pose is now (x', y', θ')

Relative motion in terms of v

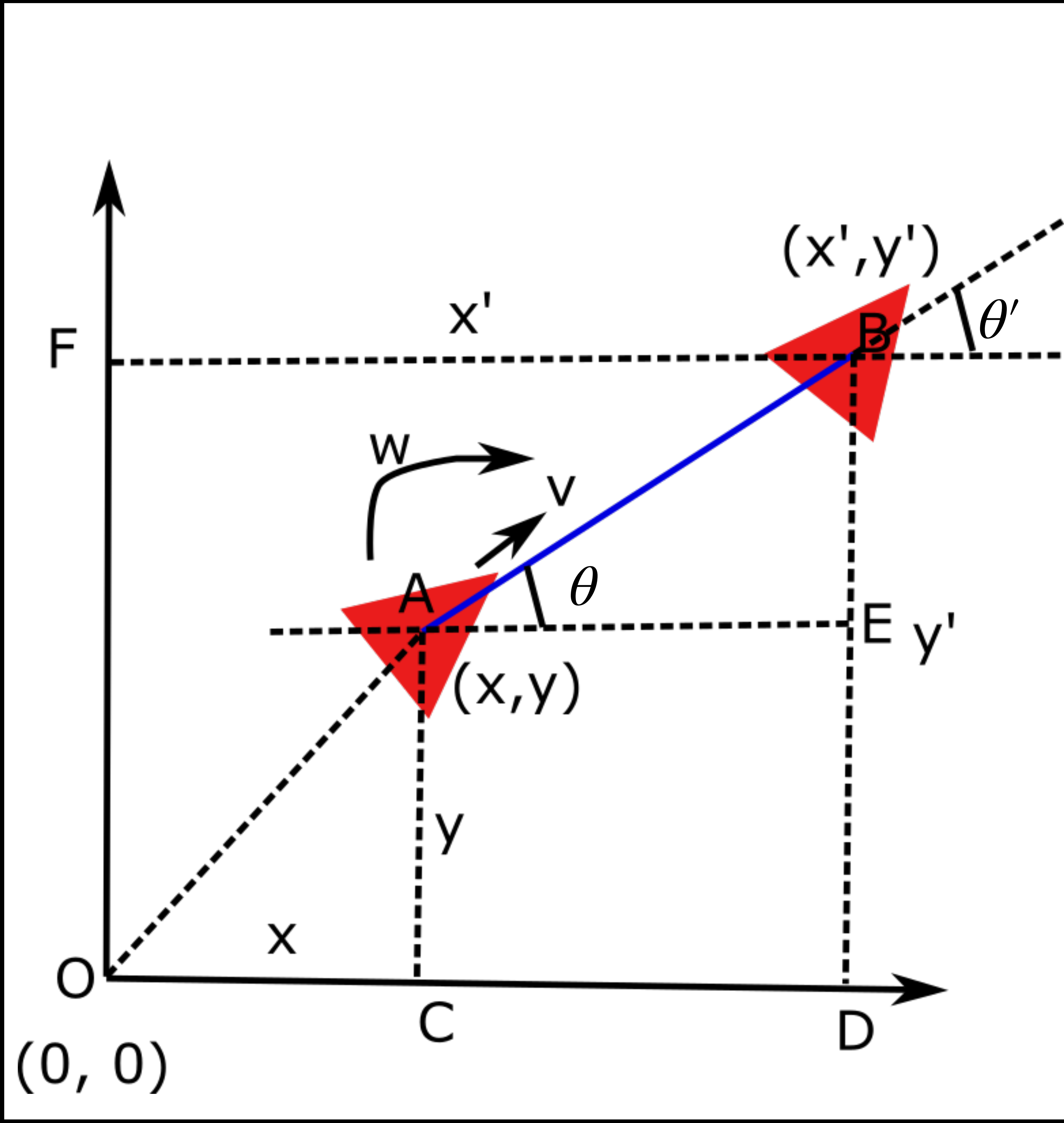
$$x' - x = v\cos\theta\, dt$$

$$y' - y = v\sin\theta\, dt$$

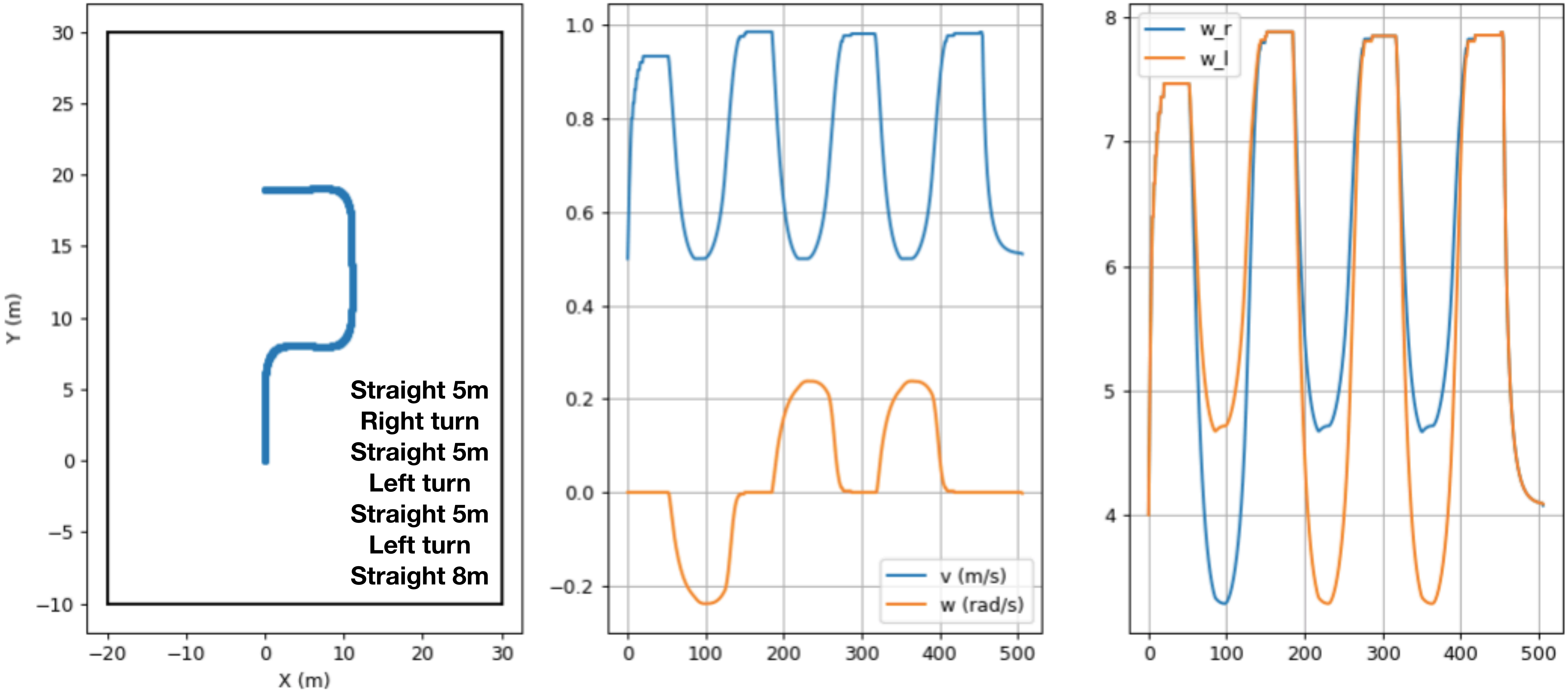
If robot also rotates at ω

$$\theta' - \theta = \omega\, dt$$

- **v and ω are commands**
- **Change in robot pose is effect**



Vehicle trajectory example



1 m/s = 3.6 kmph

1 rad/s = 9.55 rpm

Points to ponder

1. How long does it take to complete straight segments and turns?

2. Given these wheel speeds and motor speeds for peak efficiency (from data sheet in Lec 1), what else is needed mechanically?

3. For robots running at 15 Kmph, what kind of wheel speeds do we expect?

Differential-drive model

Point of contact with ground has no net force

- Wheel will get “dragged” otherwise

Wheel center moves same distance as any point in the rim (for ex: point A)

$$s = r\theta$$

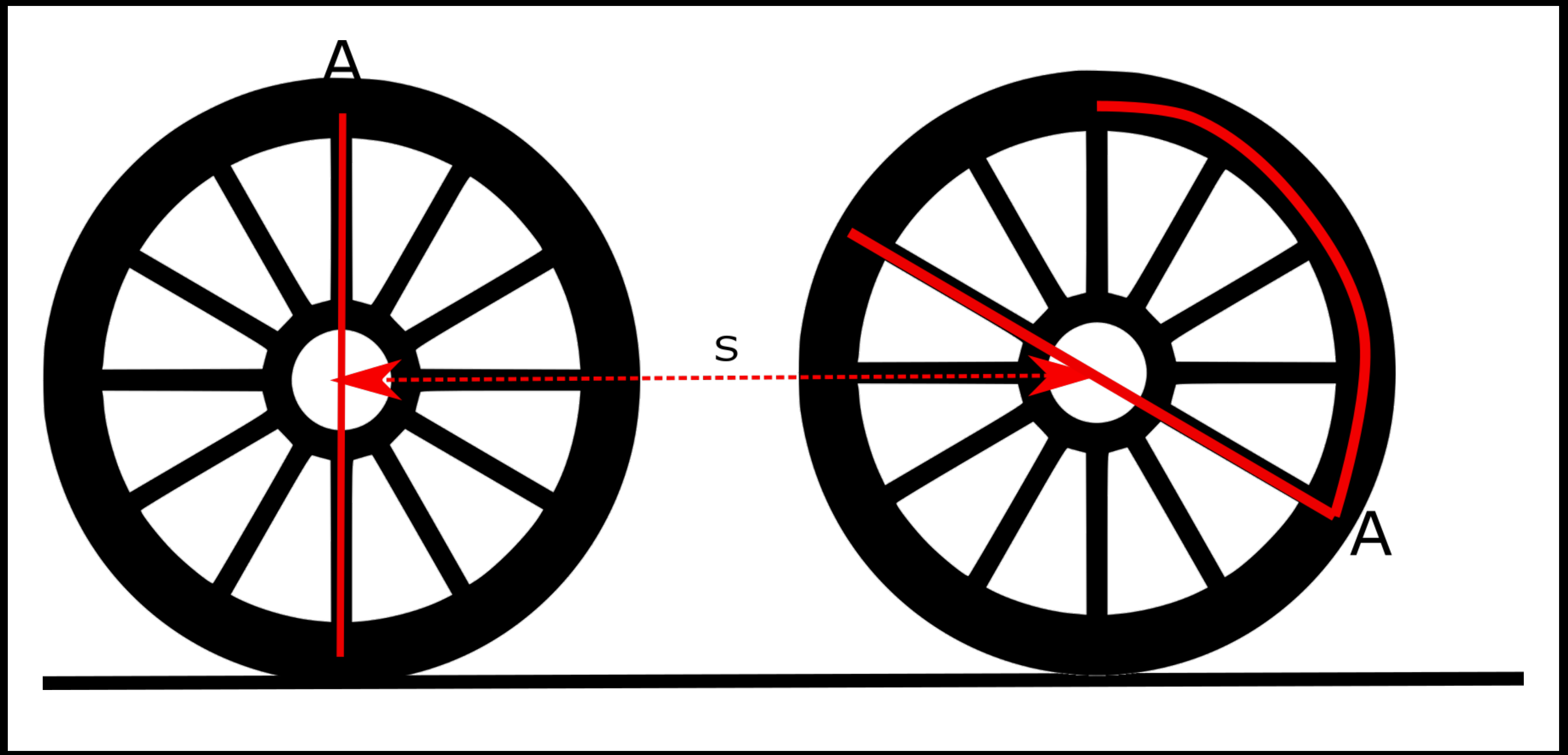
Differentiating wrt time

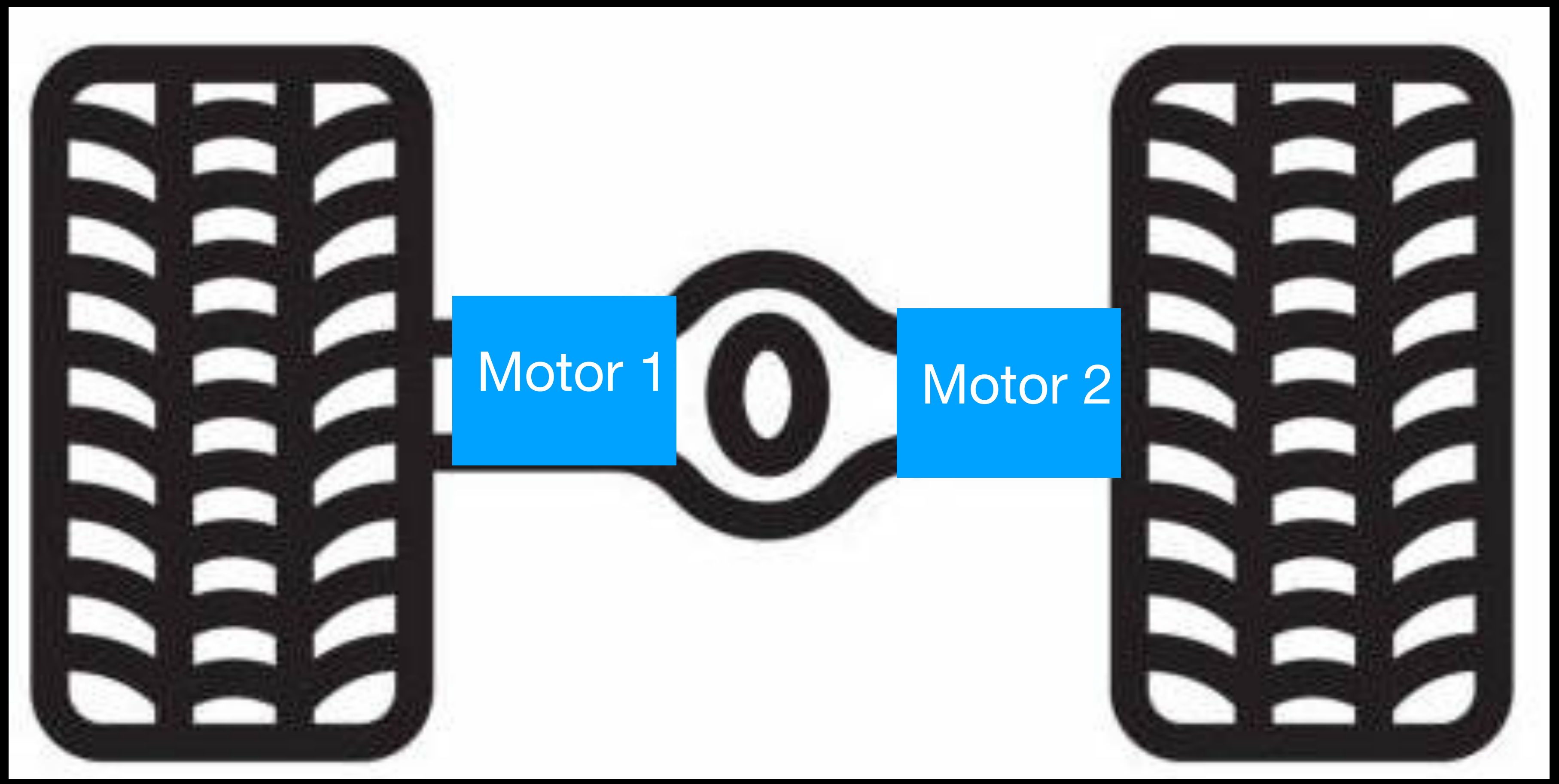
$$v = r\omega$$

Simple robot

- Connect 2 wheels through an axle
- Each wheel can be independently controlled with a motor

Roll without slipping





Differential Drive Model

Consider robot vehicle taking left turn

Two rotations happening

- 2 wheels are rotating

- Rigid body (2 wheels + axle) rotates around an imaginary point - ICC

How do they relate?

$$v_r = \omega\,(R + L/2)$$

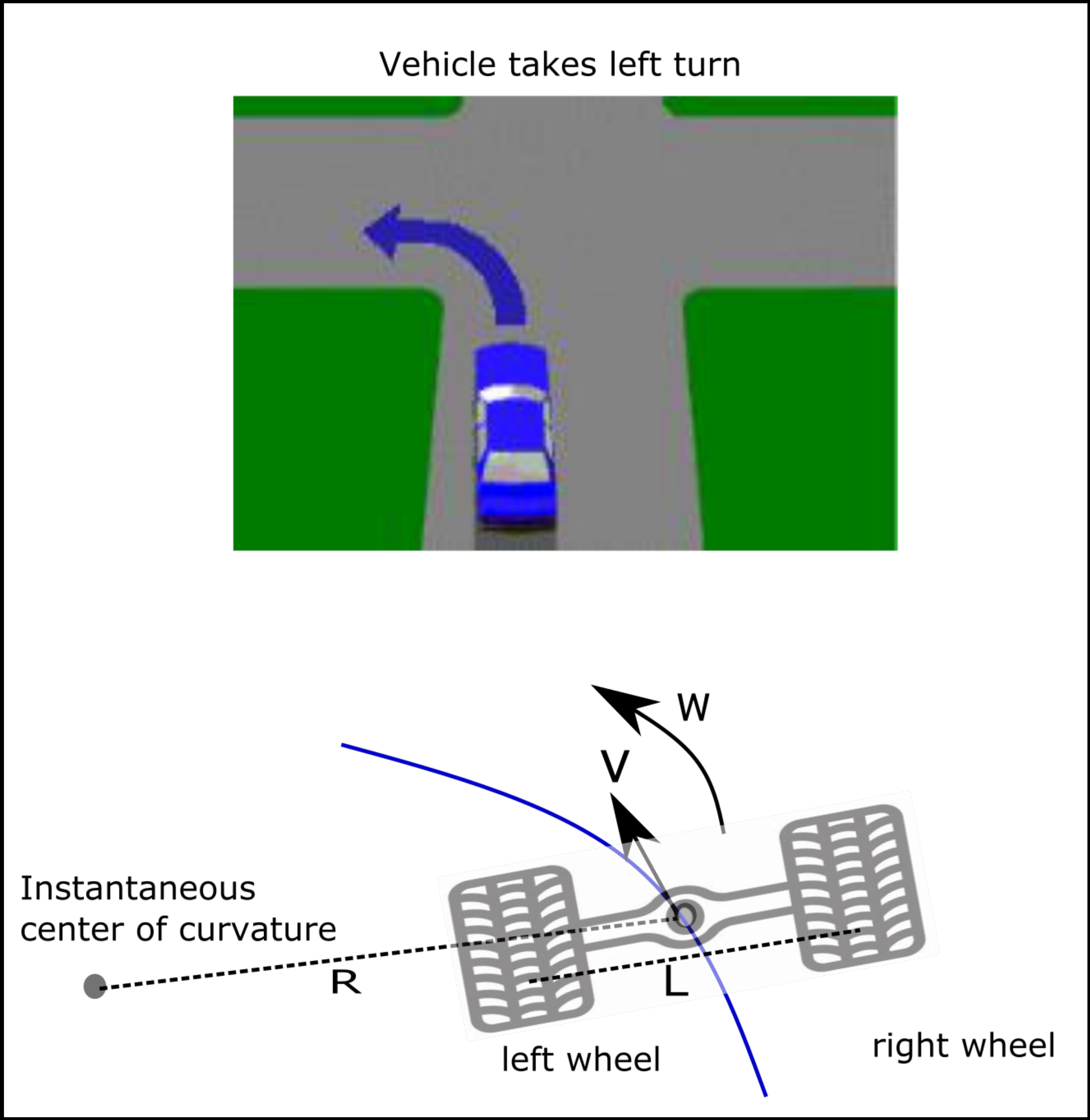
$$v_l = \omega\,(R - L/2)$$

Adding,

$(v_r + v_l)/2 = \omega\,R$ ~ avg velocity

Subtracting,

$$(v_r - v_l)/L = \omega$$



Differential drive

Each wheel has own angular velocities

$$\omega_r = v_r r \qquad \omega_l = v_l r$$

Rewriting, v and ω for the system is

$$v = \frac{r}{2}(w_r + w_l)$$

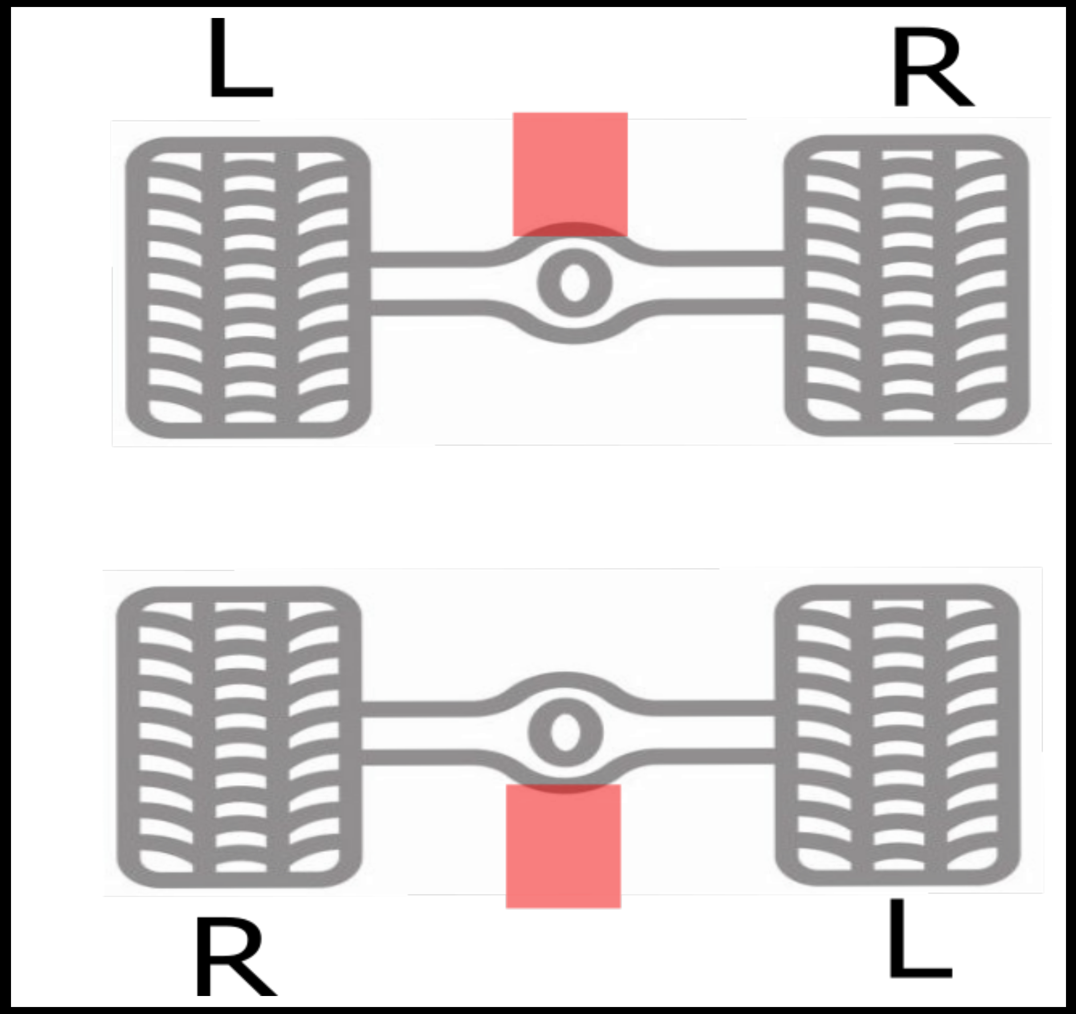
$$\omega = \frac{r}{L}(w_r - w_l)$$

Curvature

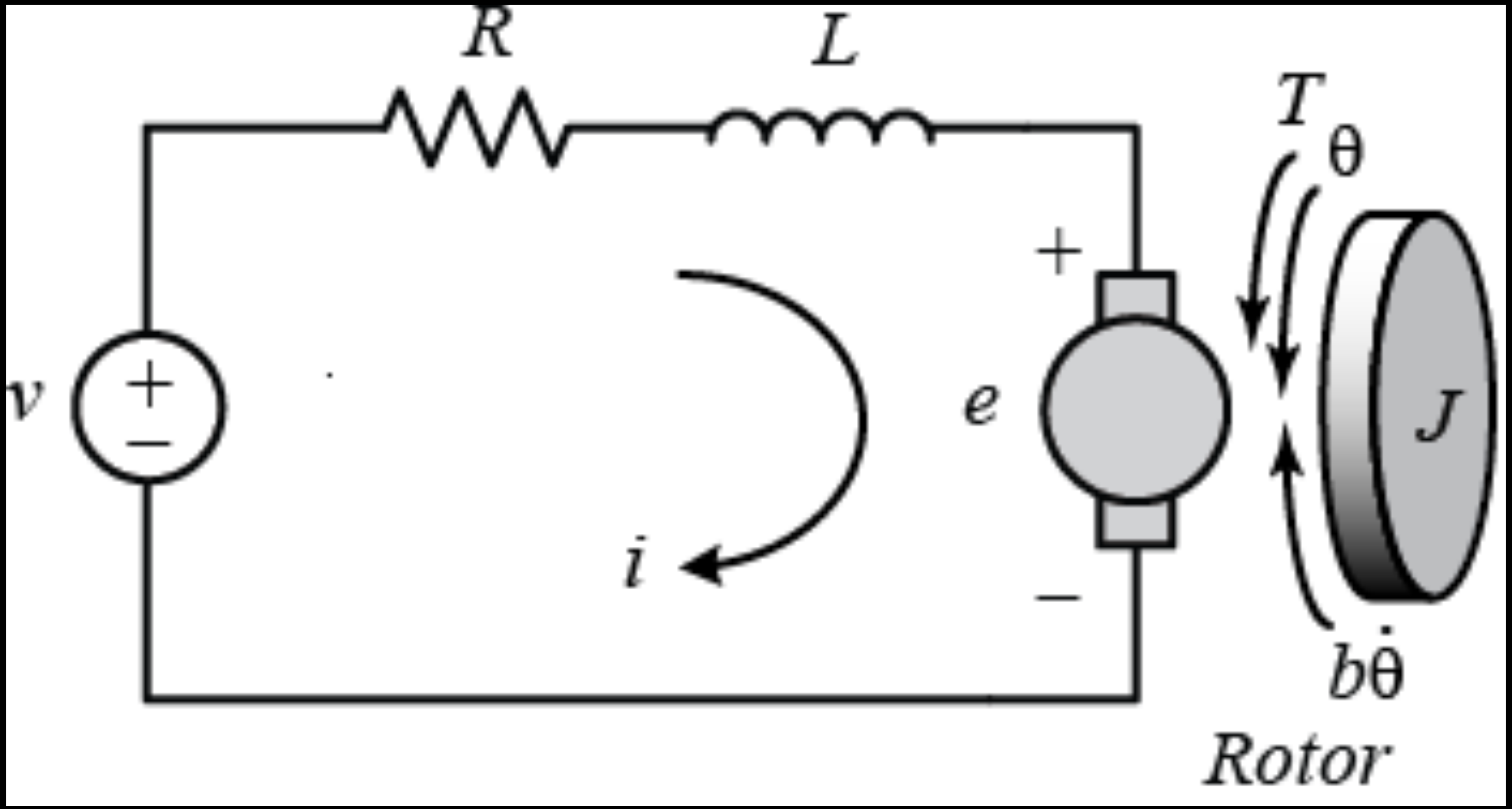
$$R = \frac{L}{2}\frac{w_r + w_l}{w_r - w_l}$$

Interesting observations:

1. $\omega_r = \omega_l \implies$ Straight line
2. $\omega_r = -\omega_l \implies$ rotates about mid-point (Inplace)



Circuit diagram for DC motor



J = M.I ~ mass
ω ~ velocity
τ ~ Force

Kirchoff's voltage law

$$L\frac{di}{dt} + Ri = V - e_b$$

Newton's 2nd law

$$J\frac{d\omega}{dt} = \tau - \tau_{load} - b\omega$$

Electro-mechanical coupling

$$e_b = k \cdot \omega$$

$$\tau = k \cdot i$$

Motor control: Magic of closed-loop feedback

Desired properties

1. Need to track reference
2. Should not be dependent on model parameters
3. Provide stable performance (Bounded-Input Bounded-Output)

Adaptive Cruise control

Simpler model problem to track vehicle velocity

Design principles equally apply to EV motor control or anywhere else

Goal: Make a vehicle run at constant reference speed v_{ref}

Input: Throttle/ brake (u)

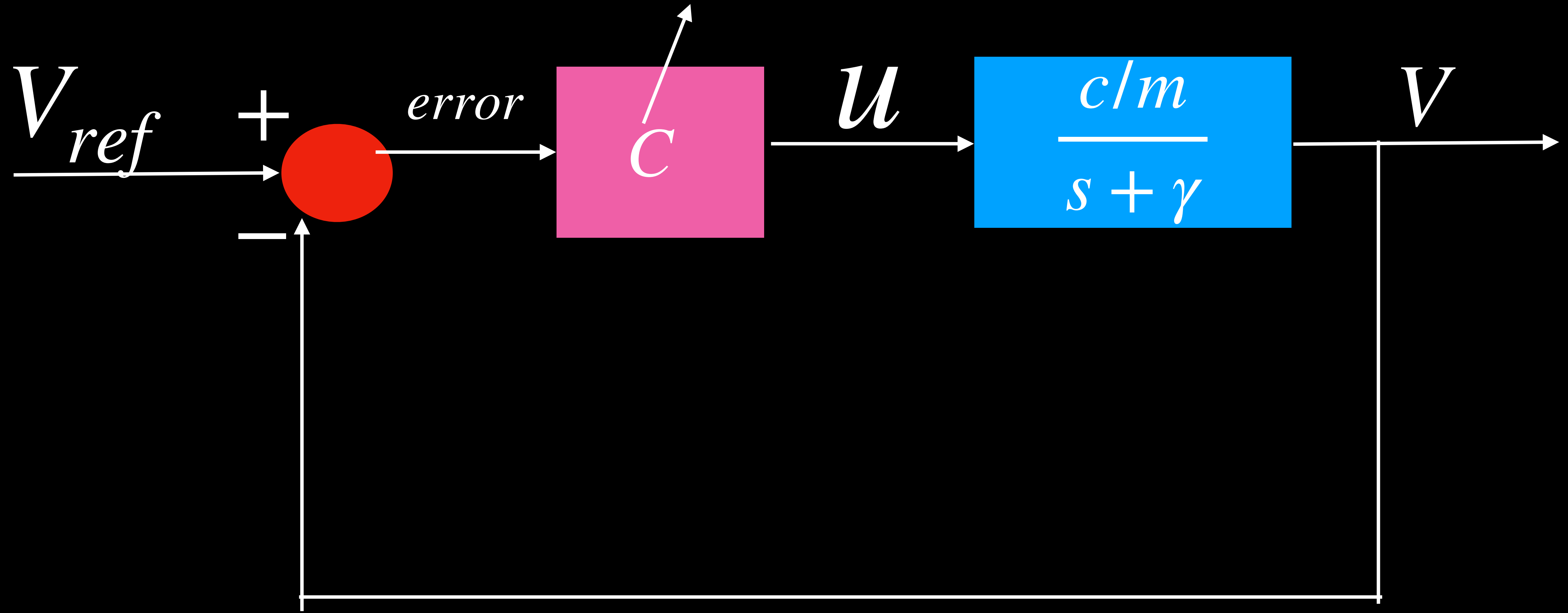
State: Velocity (v)

Newton's 2nd law: $F = ma$ or $a = \frac{1}{m}F$

Assume model: $\dot{v} = \frac{c}{m}u - \gamma v$

- c = vehicle-dependent constant
- m = mass of vehicle
- γ = air-resistance coefficient

How to design this?



Example adapted from Control of Mobile robots, Georgia tech

Adaptive Cruise control design

Three design choices

1. Bang-Bang control

If error is positive, hit max throttle or else hit max brake

$$\dot{v} = \frac{c}{m}k(v_{ref} - v) - \gamma v$$

2. Proportional control

Control is proportional to error

Steady-state $\implies \dot{v} = 0$

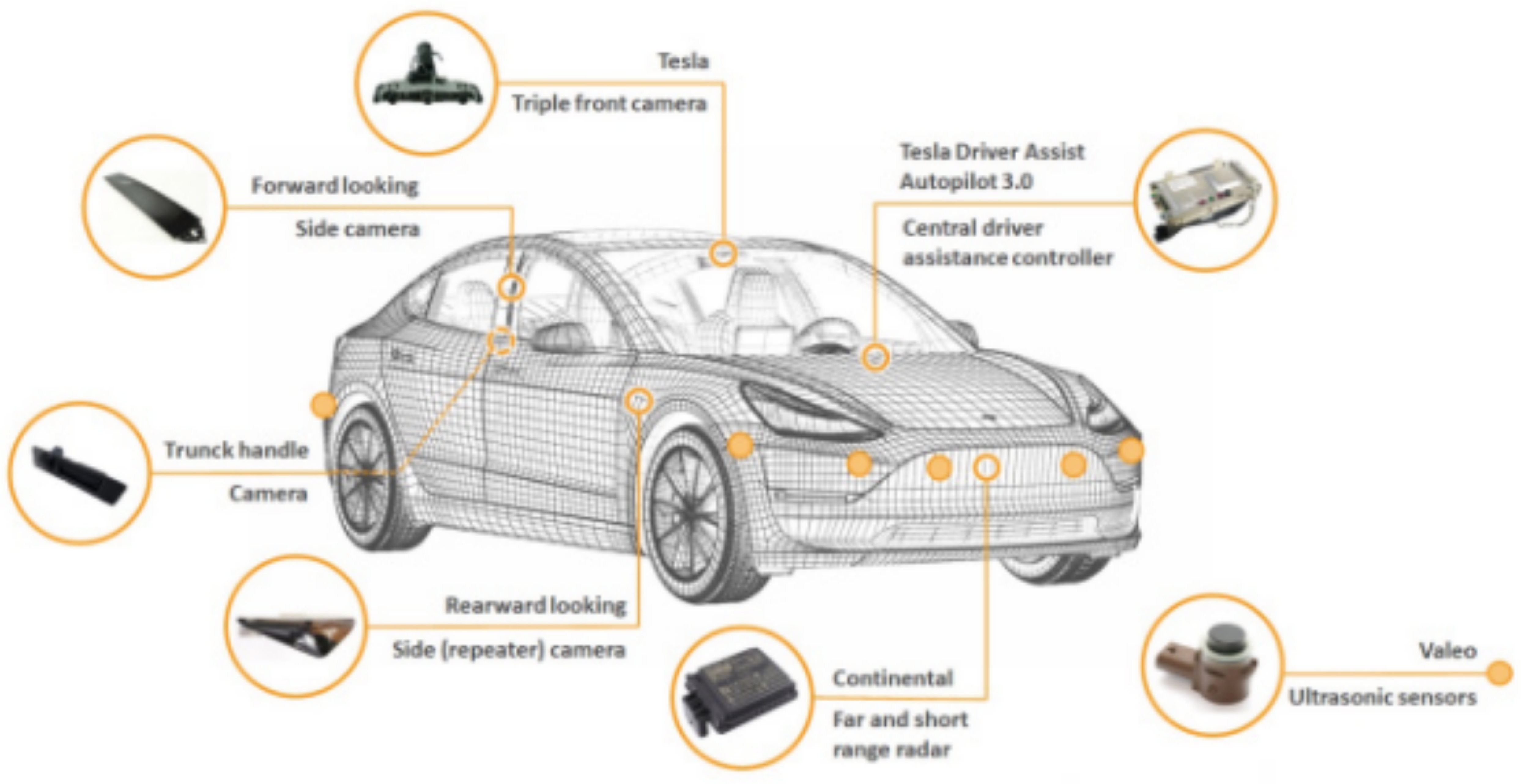
$$v_{ss} = \frac{ck}{ck + m\gamma}v_{ref}$$

3. PI control

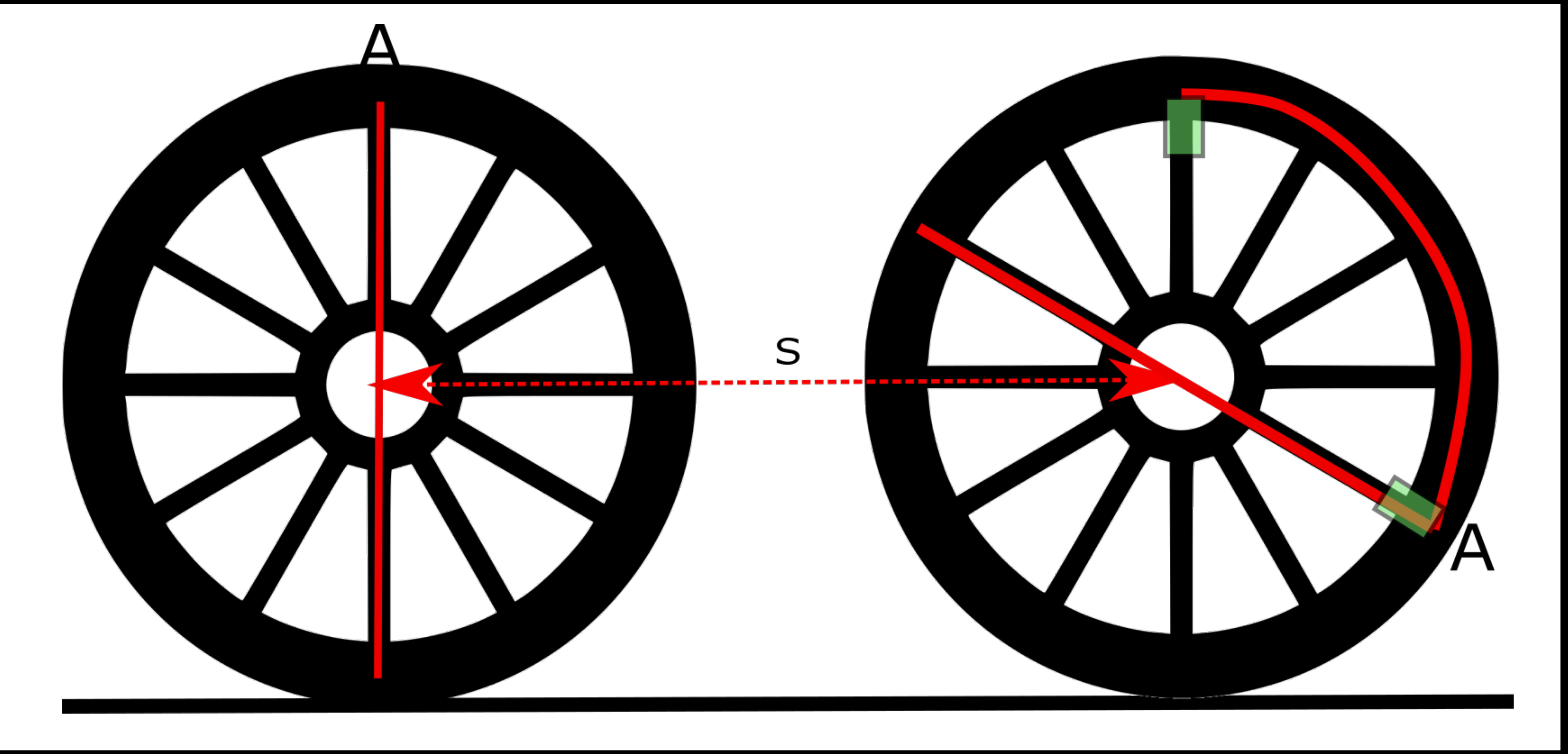
Control is proportional to both current error as well as accumulation of previous error terms

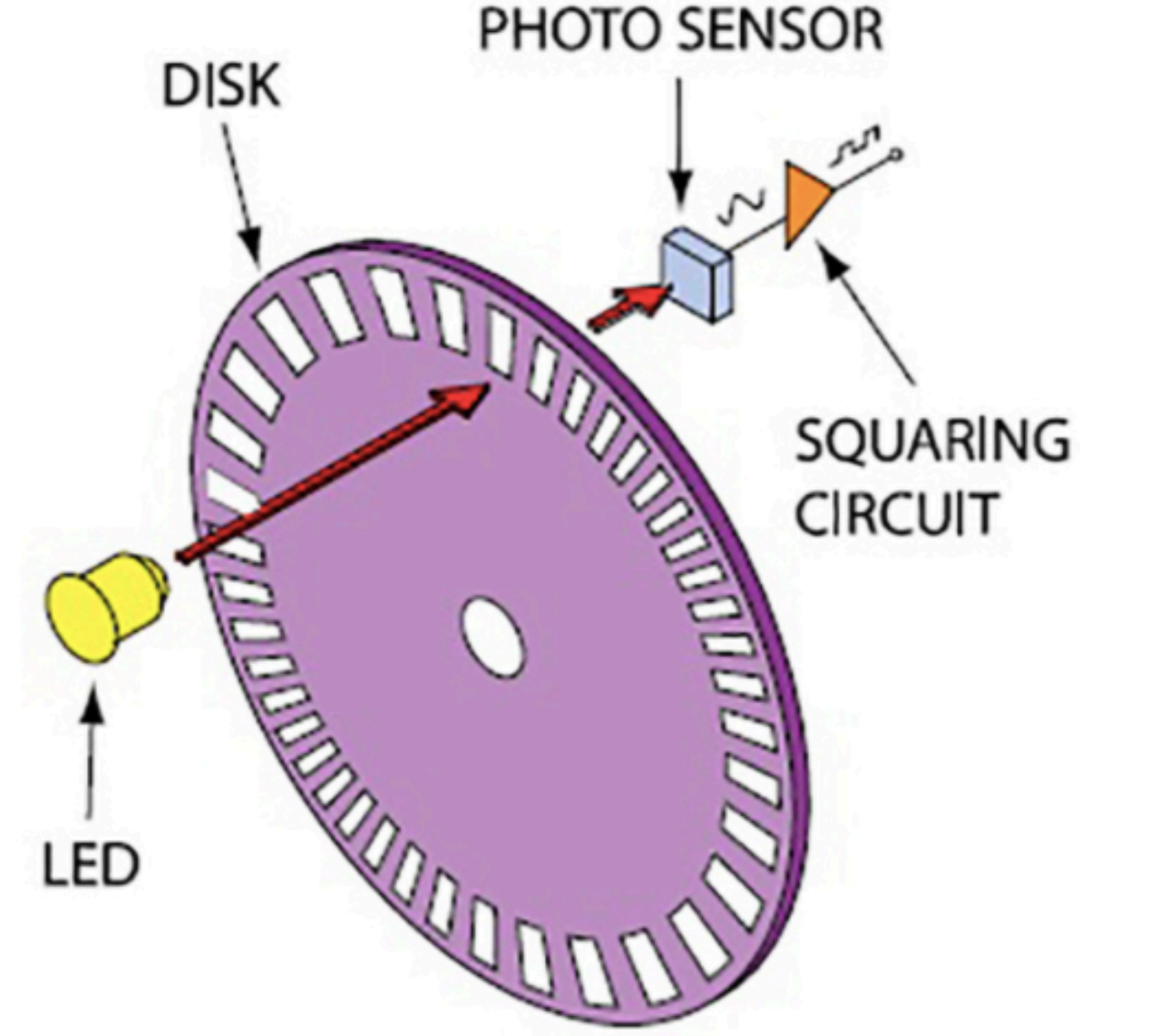
Illustrations in Jupiter notebook

Sensor placements in Model 3



Wheel encoders



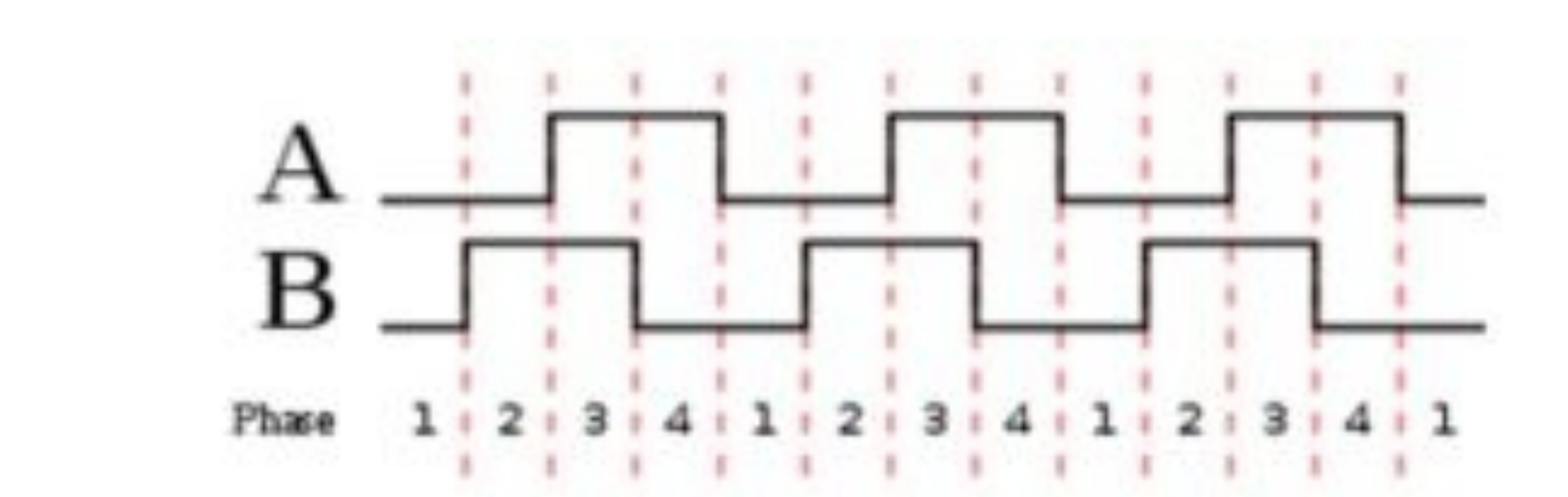


Every T ms

Encoder reports "ticks" wheel moved = n

Encoder resolution = N

Distance moved = $2\pi R \frac{n}{N}$



Source: Anaheim Automation

Wheel encoders

Usage

- Smallest movement that can be measured is $\dfrac{2\pi R}{N}$

- Largest speed that can be measured is $\dfrac{2\pi R}{dt}$

Pros:

- Fairly accurate estimates of linear/ angular velocity
- Distances and rotations are accurate in short-term

Cons:

- Vehicle position “drifts” when v, ω is integrated over longer periods

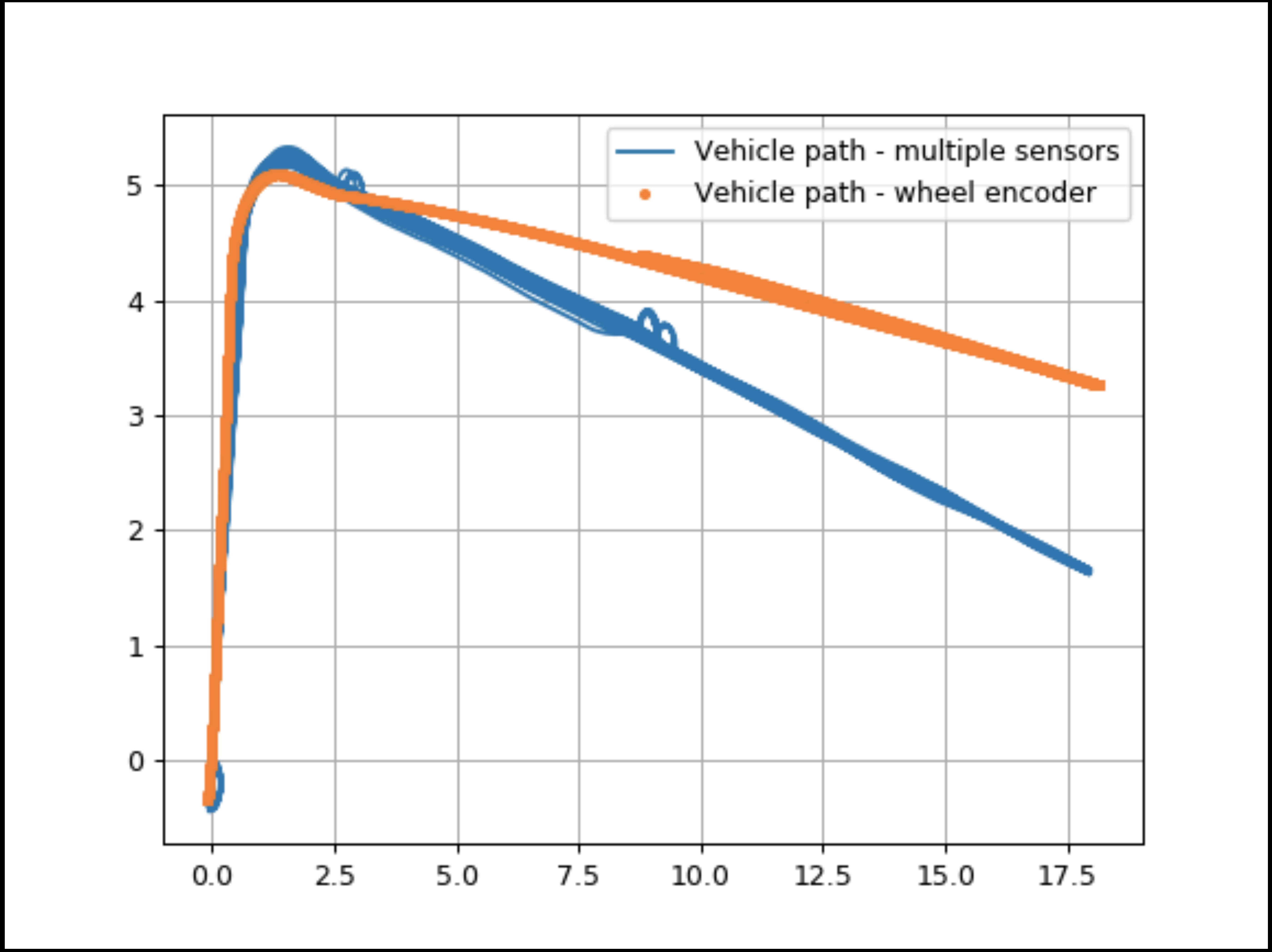


Illustration in Jupiter notebook

Lidar

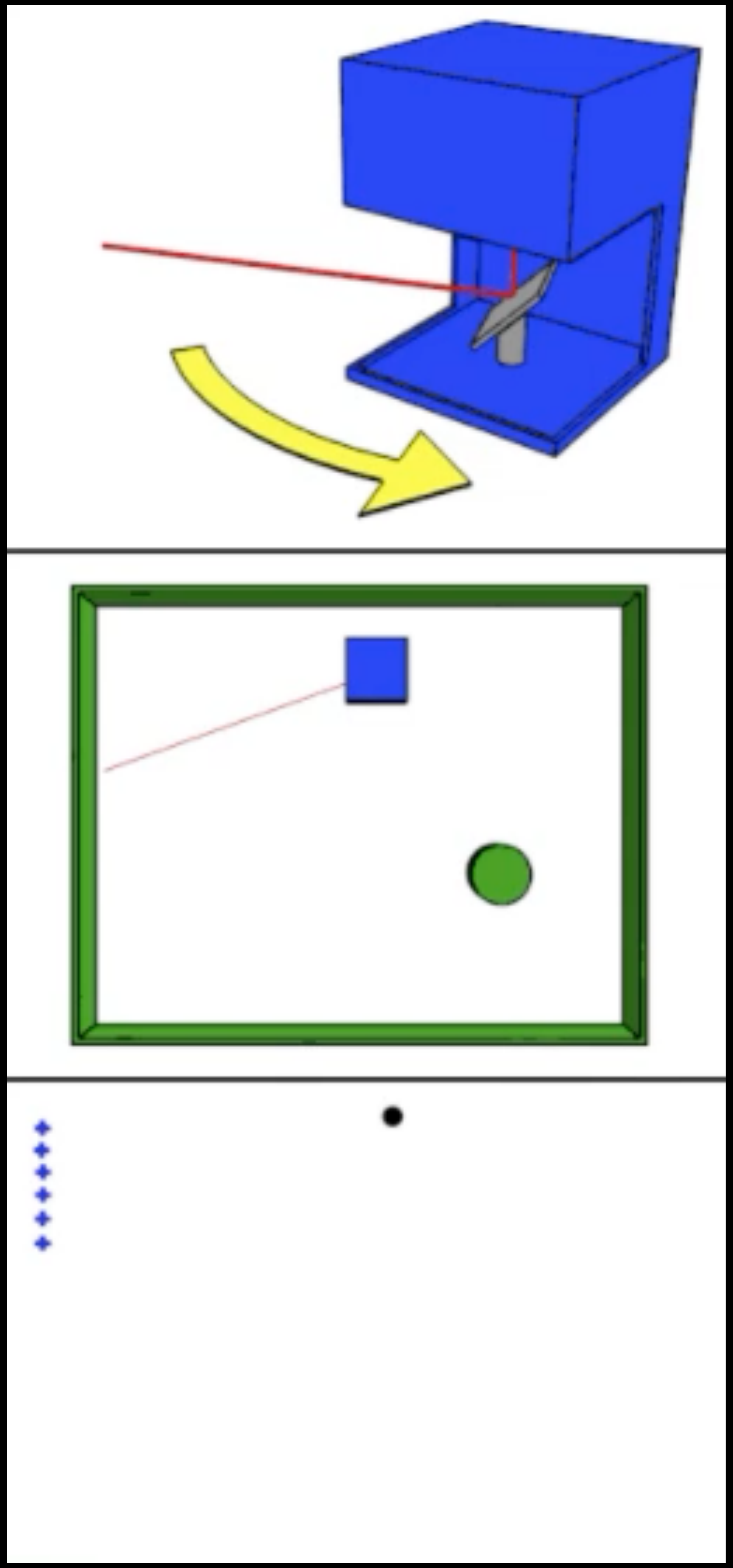
Light detection and ranging

Visible or Near-IR light used to image objects

Object distance calculated from time-of-flight estimates

Several advantages

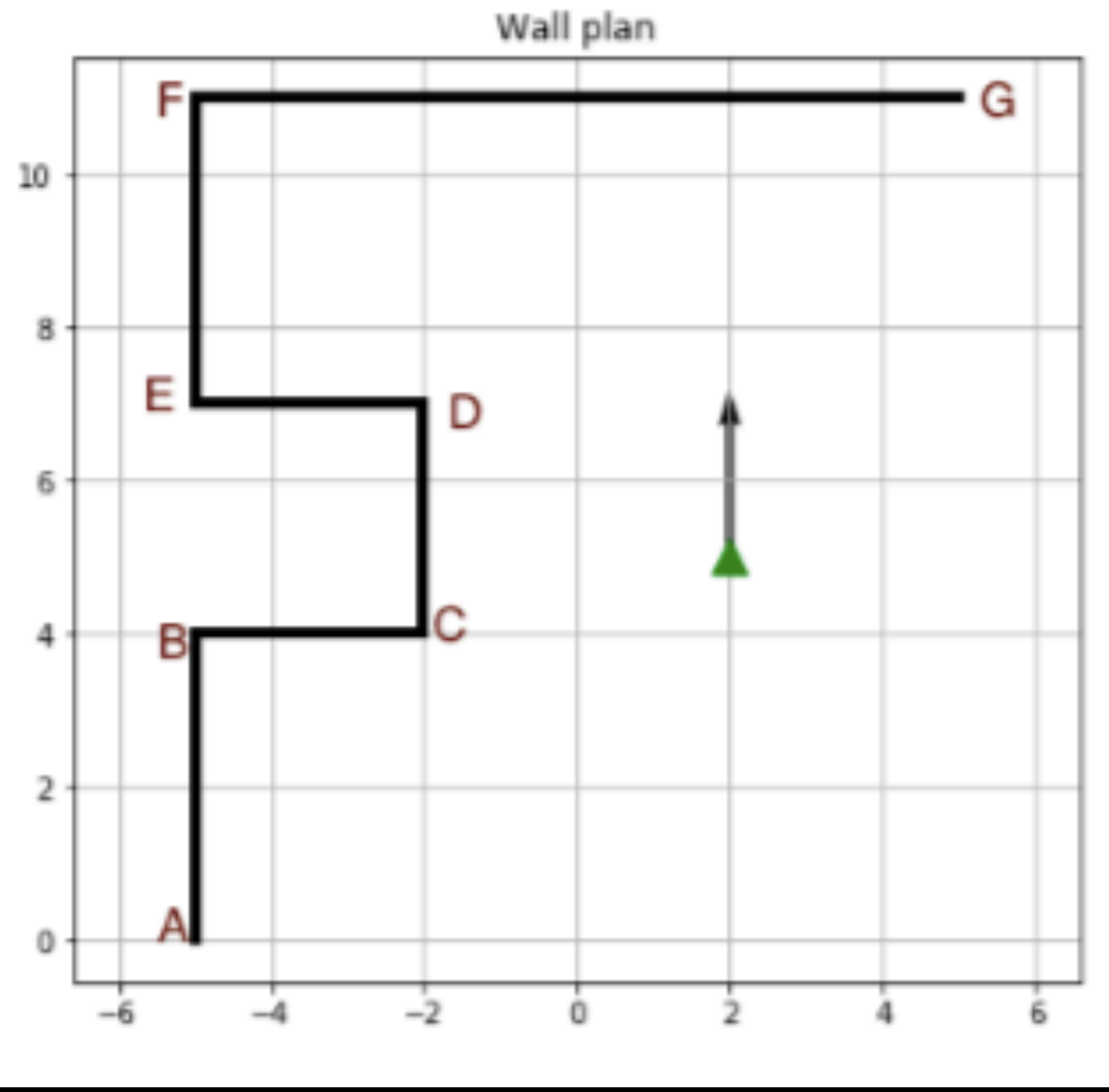
- Low Noise floor
- Independent of ambient lighting
- “Ready-made” 3-D imaging

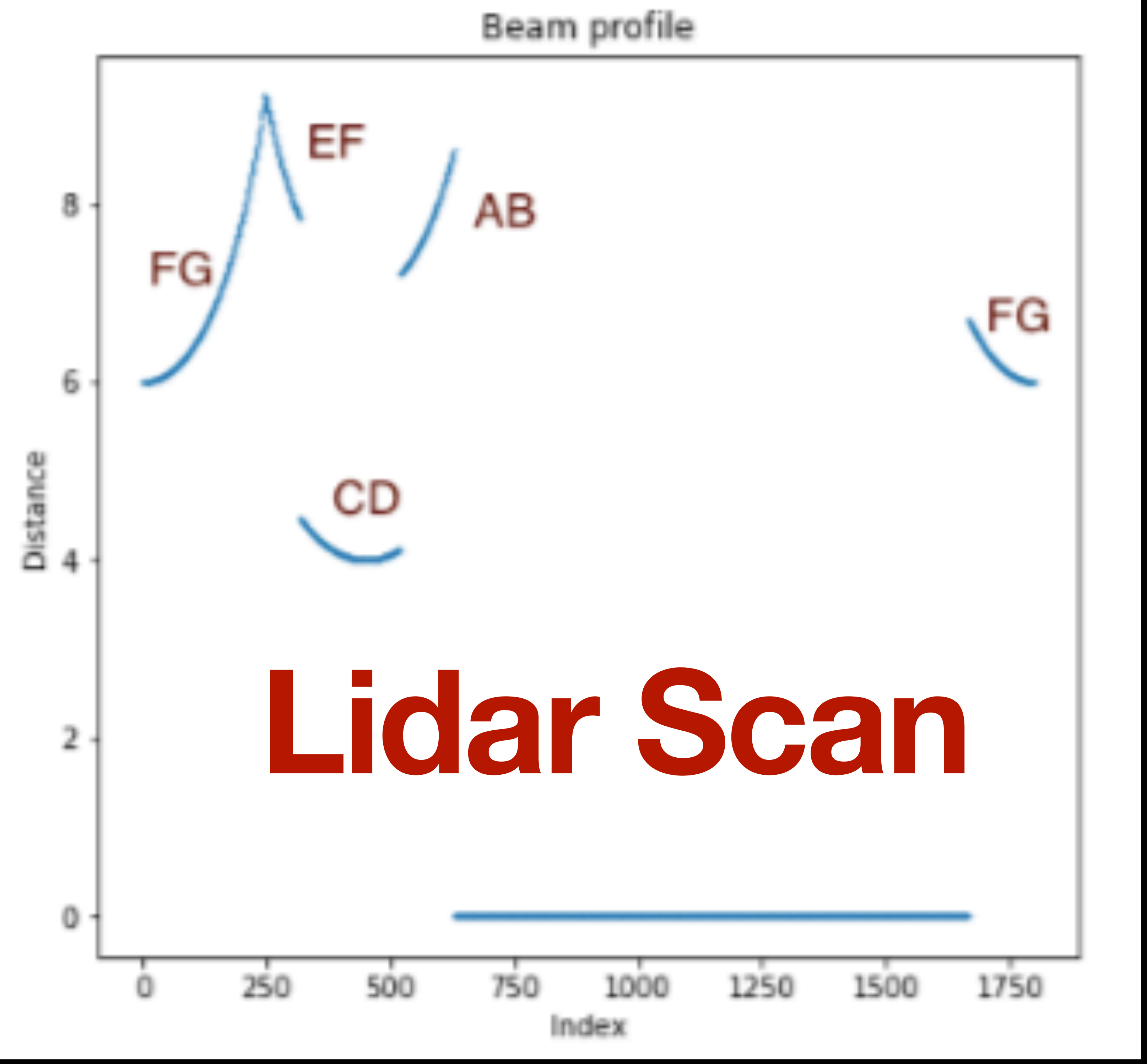


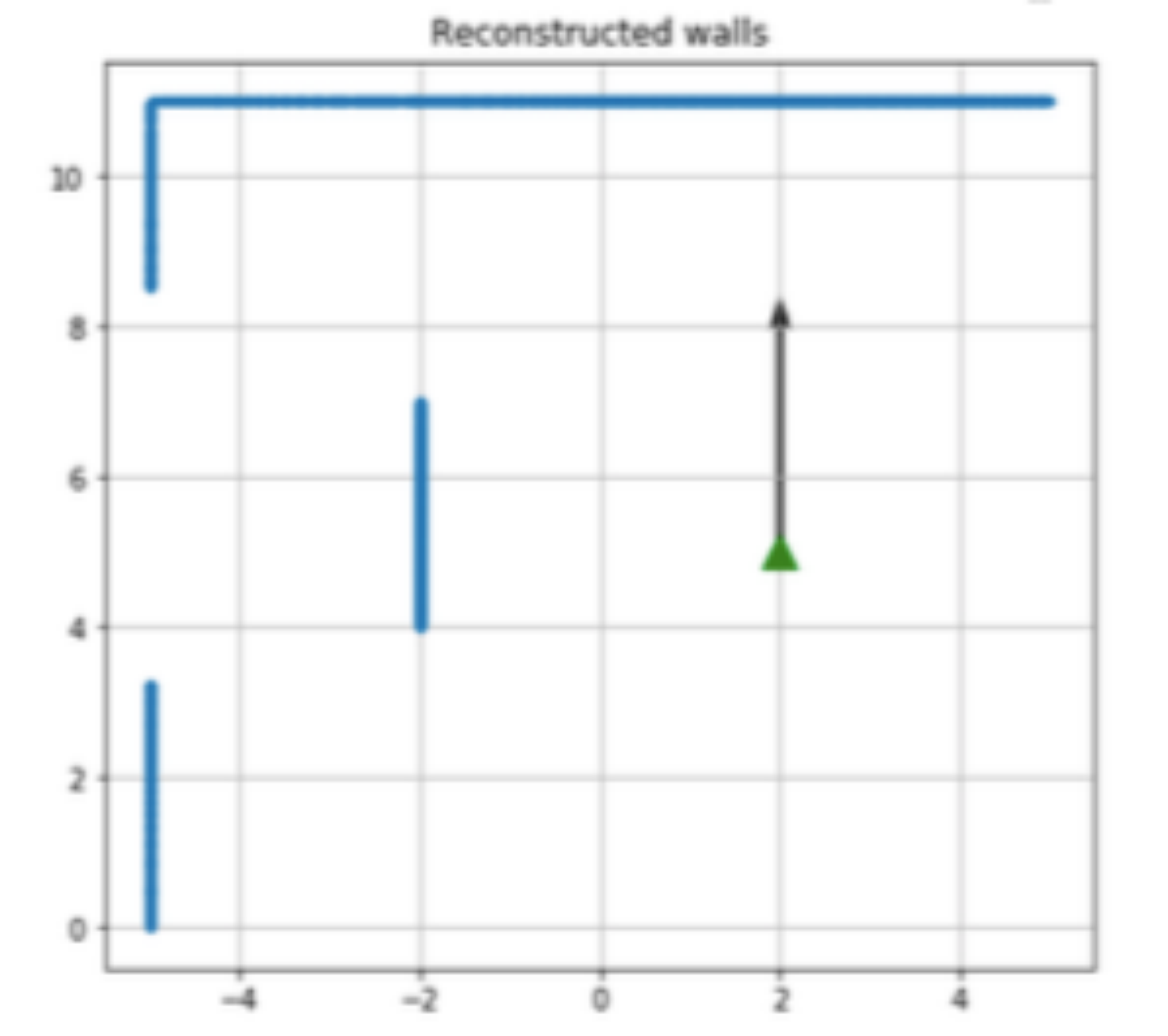
Video: https://en.wikipedia.org/wiki/Lidar

2-D Lidar

Single Laser beam rotating 360 degrees
Return reflections d recorded along with azimuth angle θ

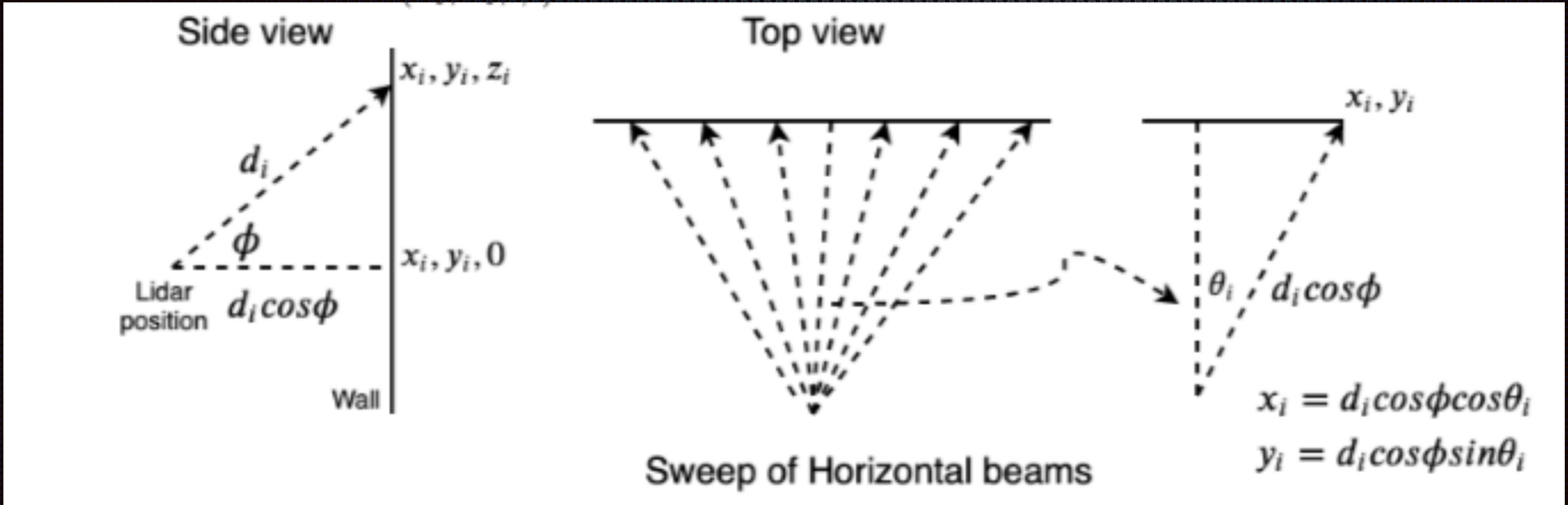






2-D Lidar simulator animation in Jupiter notebook

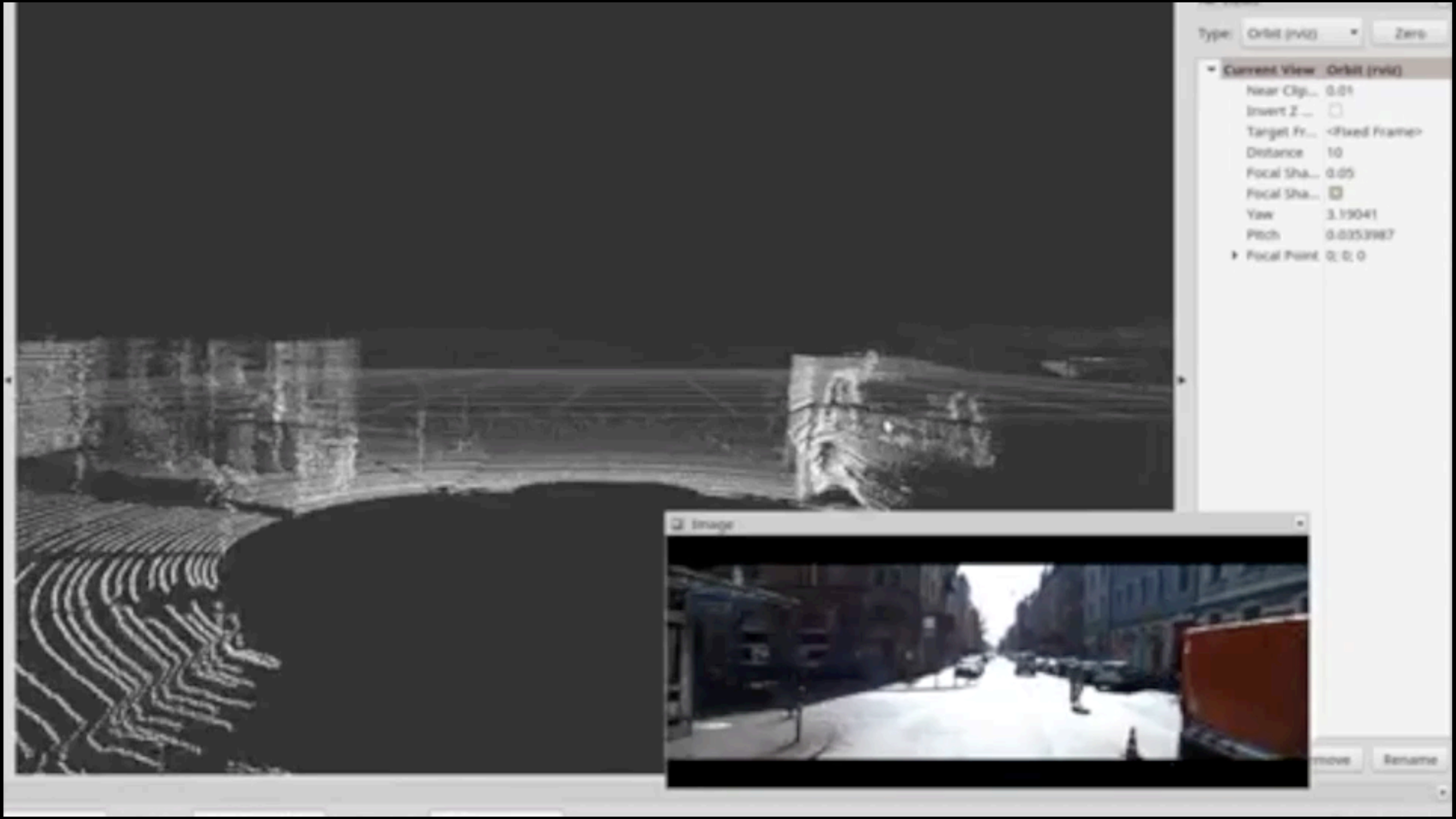
3-D Lidar



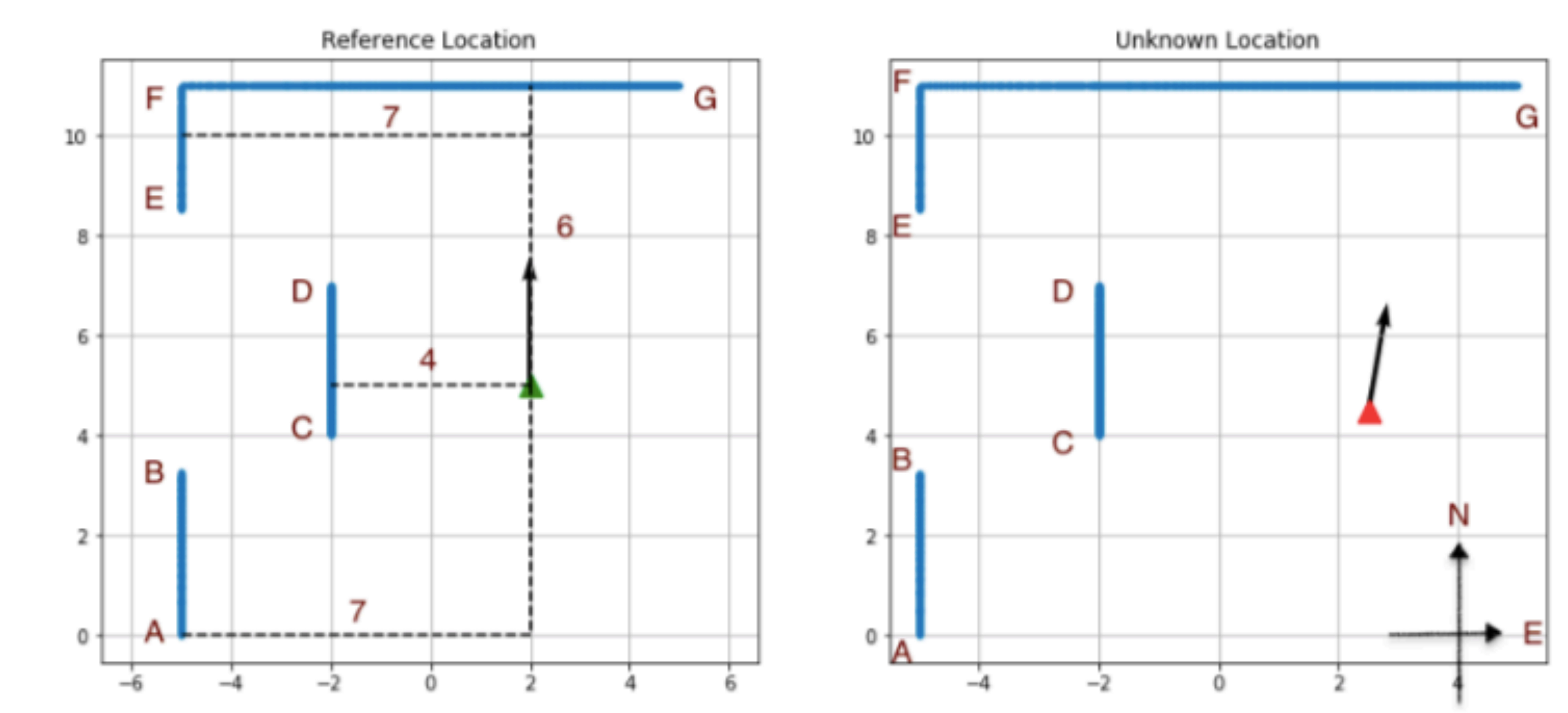
N-laser beams rotating 360 degrees
Span different elevation angles ϕ

Object which reflects is at (d, θ, ϕ)
Can be converted to cartesian coordinates (x, y, z)

3-D Lidar visualization



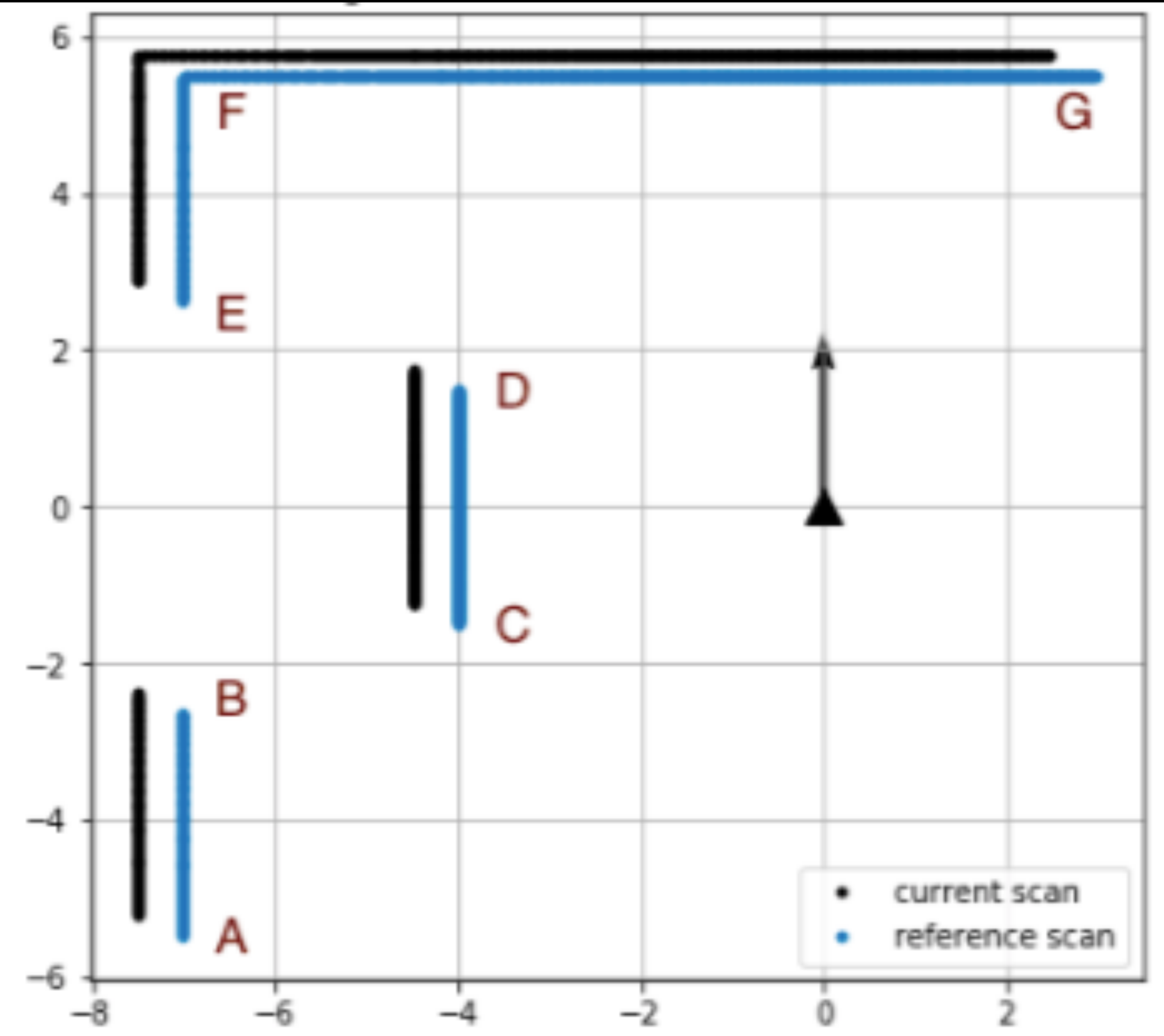
Localization



Scan-Matching

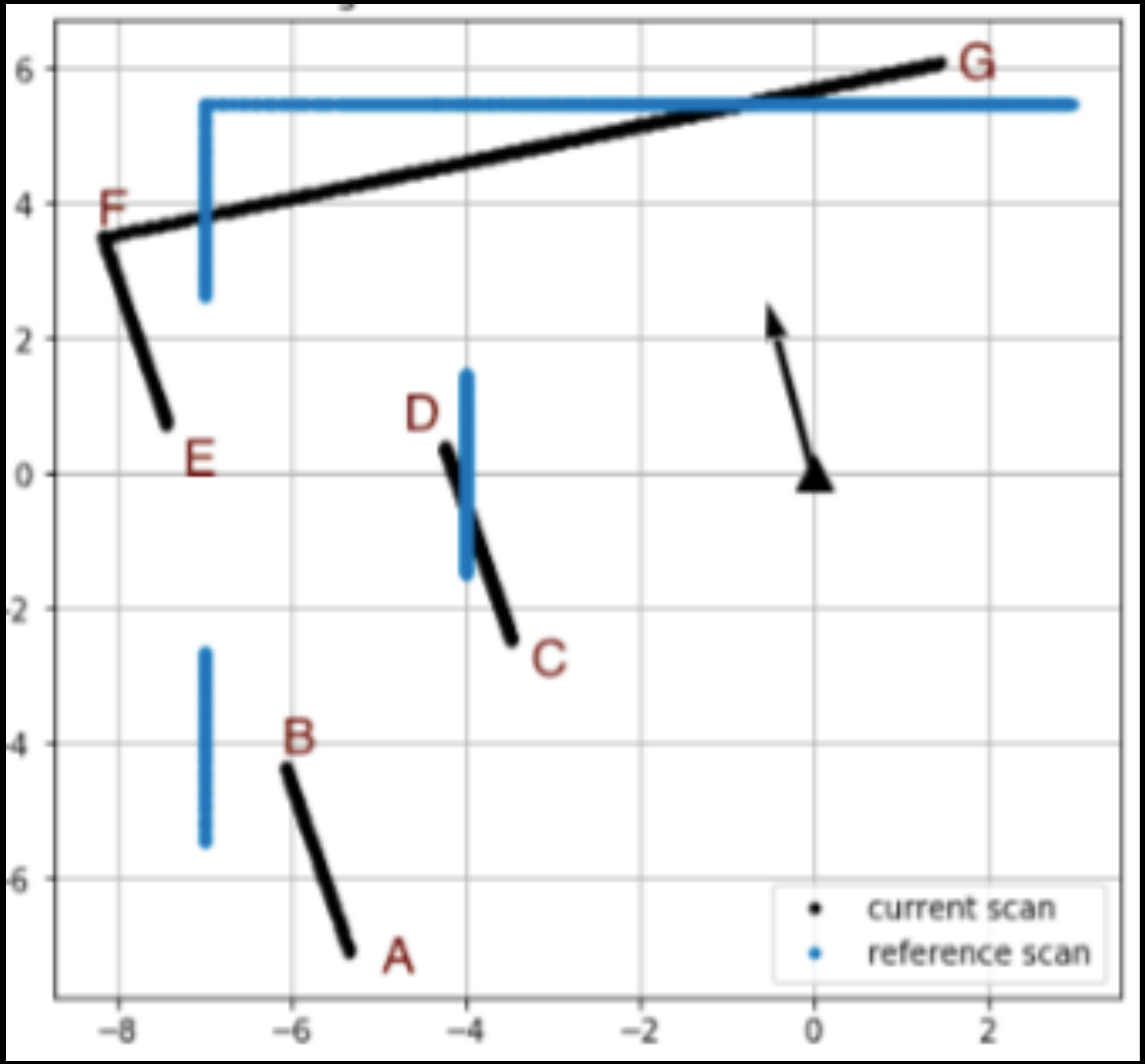
- Wall positions relative to “green” robot position known
- Can the “red” robot position be determined? Yes
    - Lidar scan at red position is made available
    - Distance/Orientation between green and red positions is not widely different

Pure translation



- Blue walls - observed when robot is at (x_r, y_r)
- Black walls - observed when $(x_r + 0.5, y_r - 0.25)$
- Walls EF, CD, AB have moved 0.5m away
- Wall FG has moved 0.25m away

Pure rotation



- Vehicle stays at (x_r, y_r)
- There is a rotation of 15 degrees
- Current observation
- All 4 walls have rotated counter-clockwise 15 degrees

Iterative Closest point

- Two sets of points P_1 and P_2
- Each set has multiple 2-tuples (x, y) indicating position
- Pure translation

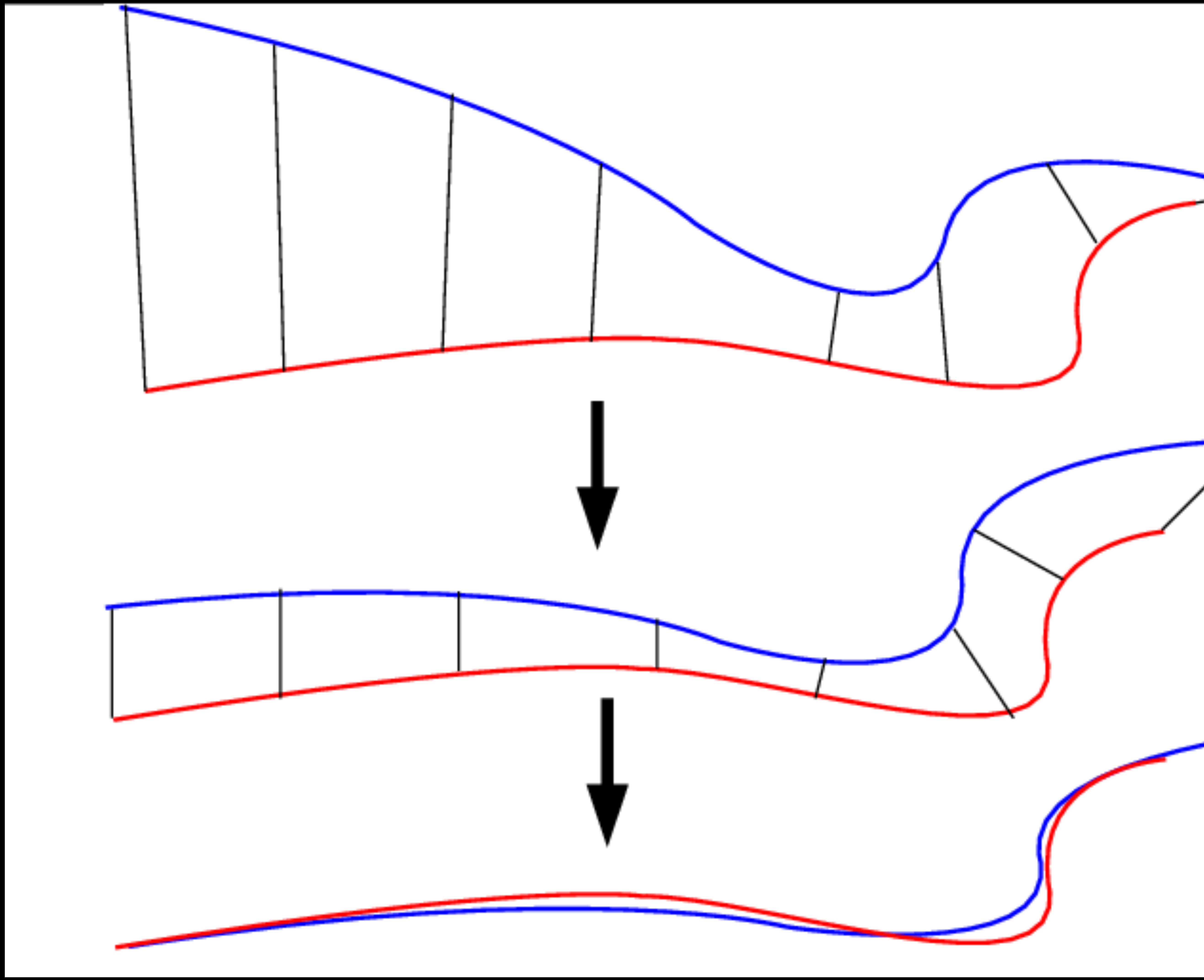
$$P_2 = P_1 + t$$

- Pure rotation

$$P_2 = RP_1$$

- ICP

$$\min_{R,t} ||R * P_1 + t - P_2||^2$$



Sensor fusion

Lidar measurements still have noise (all sensors do!)

Scan matches are not perfect

- Dynamic changes (occlusion)
- “Bad” static objects (Glass, mesh, reflective surfaces)
- Sparsity of scatterers

Wheel encoders

- Smooth outputs in the short-term
- Drifts over long periods of time

Lidar

- Noisier output
- Stable over long runs

Kalman filter - combine the pose estimates from multiple sensors

Descriptive Statistics

Random variables, Mean, Variance, Gaussian distribution

- Random Variables
  - Bit of a misnomer
  - Function maps outcomes of a random experiment to real-values
- Say, X is the random variable
- Coin toss => $X \in$ {Heads, Tails}
- Die thrown => $X \in \{1, 2, 3, 4, 5, 6\}$
- Political party of choice in a survey => $X \in$ {Party A, Party B, Party C, … }
- Temperature of people entering an office
  - Continuous
  - $X \in [96, 103]$
- **Exercises in Python notebook**

1-D Kalman filter

Two thermometers have following standard deviations

Therm 1: 3 deg C

Therm 2: 1 deg C

[Calibration could have been done by measuring and comparing against known objects]

One particular person was measured 101 and 98 on Therm 1 and Therm 2

a. Does he have fever? How to combine the 2 readings?

b. Is the combined reading “better” than the individual reading?

$\sigma_1^2 = 9$ and $\sigma_2^2 = 1$

$$\hat{T} = \frac{\sigma_2^2}{\sigma_1^2 + \sigma_2^2} T_1 + \frac{\sigma_1^2}{\sigma_1^2 + \sigma_2^2} T_2$$

$$\hat{T} = 0.1 * 101 + 0.9 * 98 = 98.3$$

$$\hat{\sigma}^2 = \frac{\sigma_1^2 \sigma_2^2}{\sigma_1^2 + \sigma_2^2}$$

$$\hat{\sigma}^2 = 0.9$$

Better than a single measurement!