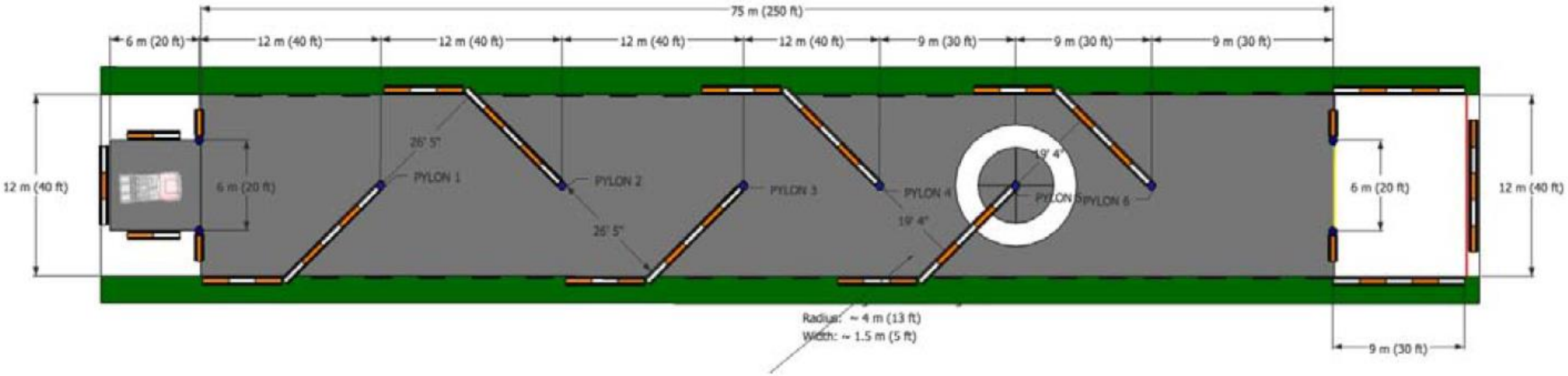
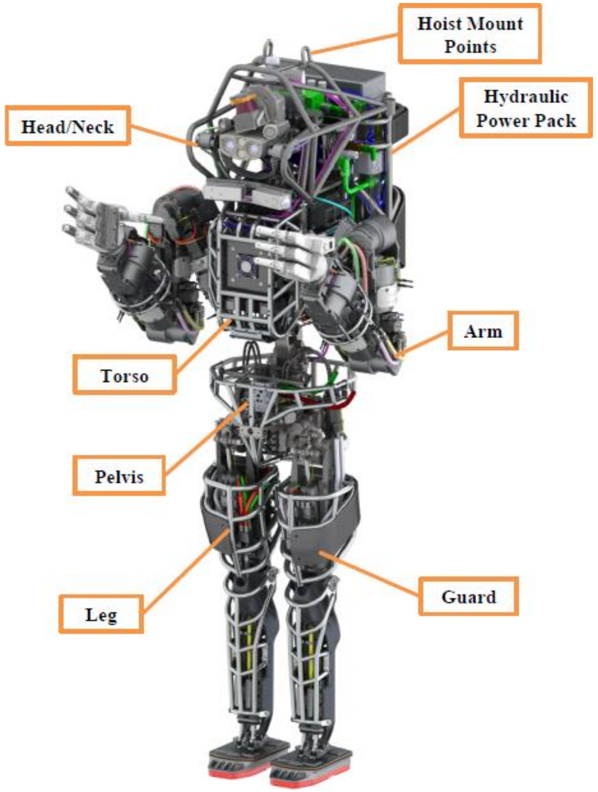
Modeling and Control

12 Feb

DARPA Robotics Challenge

- 4 points task
  - Robot drives the vehicle through the course (1)
  - Robot gets out of the vehicle and travels dismounted out of the end zone (2)
  - Bonus point (1)



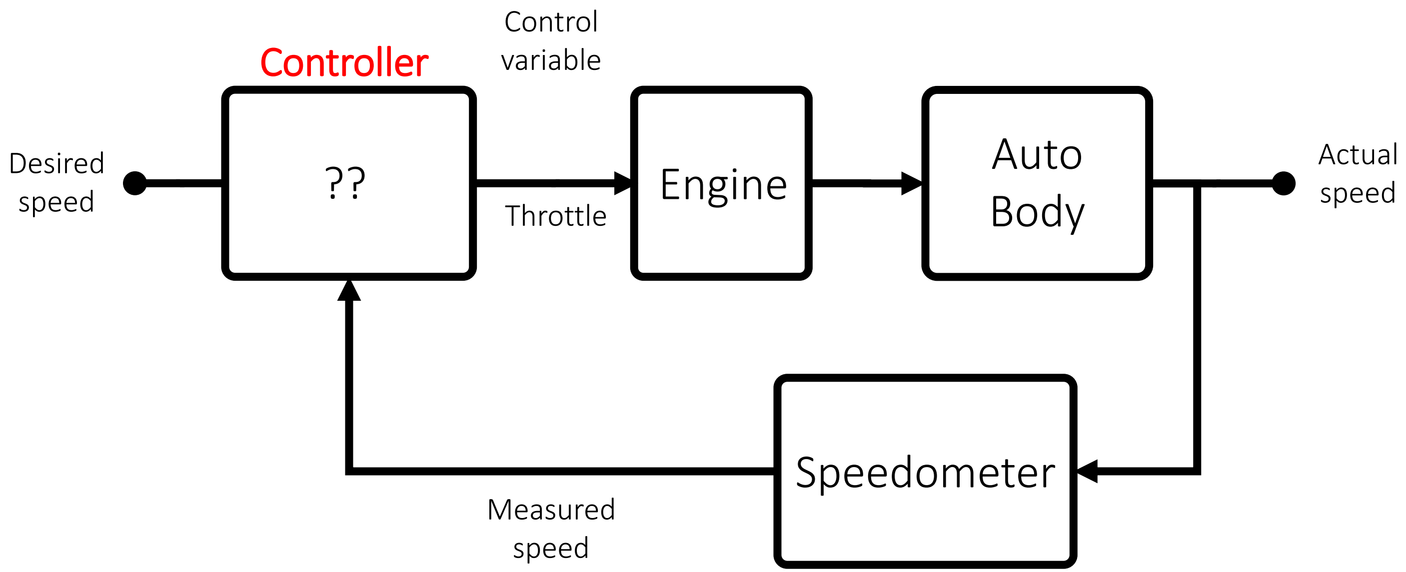



POLARIS
2

Control?

The process of causing a system variable to conform to some desired value

Example: Cruise Control



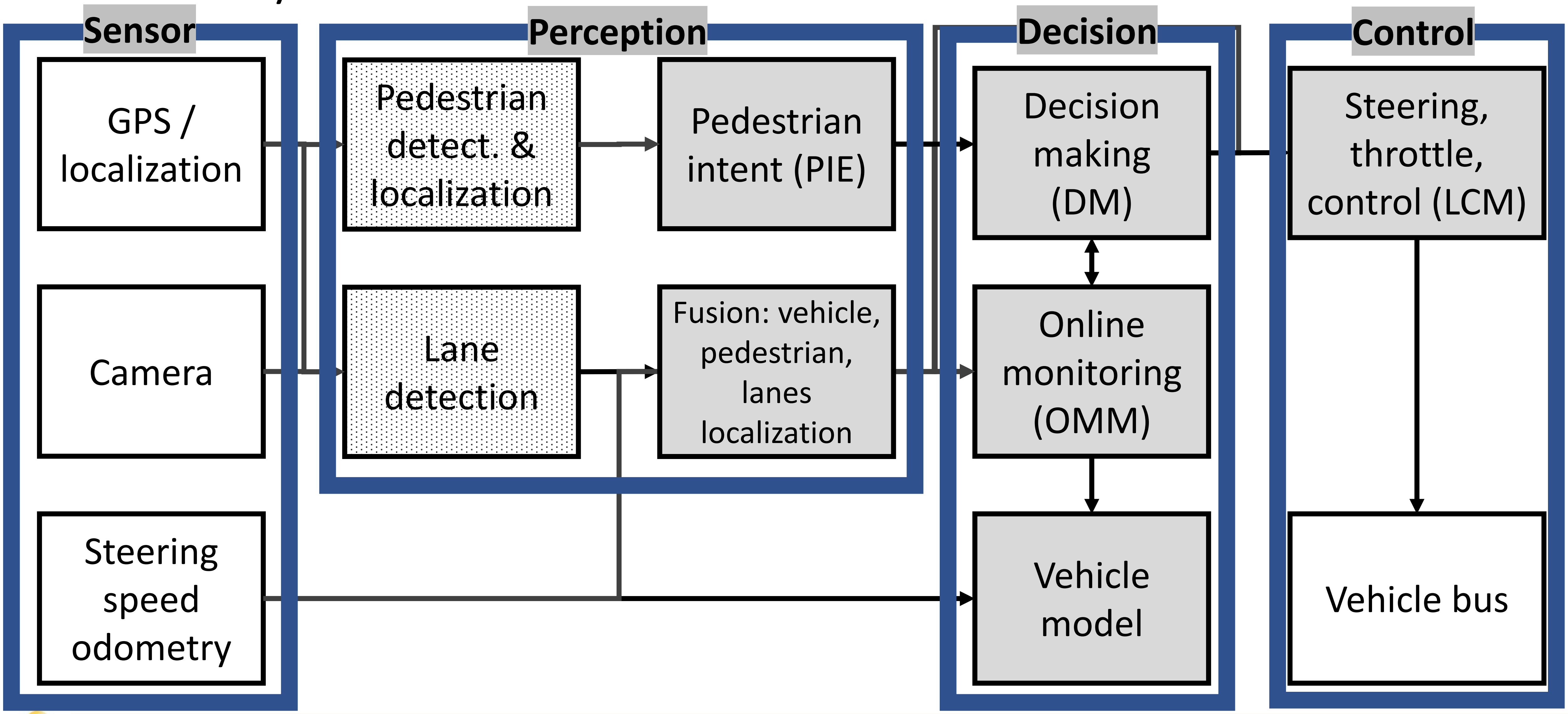
What do we want to control?

-> Car position and orientation

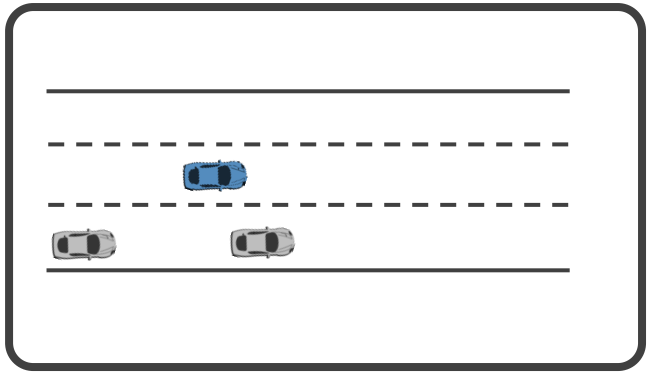
What can we use?

-> Steering wheel and pedal(s)

GEM system


Sensor
GPS / localization
Camera
Steering speed odometry
Perception
Pedestrian detect. & localization
Pedestrian intent (PIE)
Lane detection
Fusion: vehicle, pedestrian, lanes localization
Decision
Decision making (DM)
Online monitoring (OMM)
Vehicle model
Control
Steering, throttle, control (LCM)
Vehicle bus

Dynamical system models



Nonlinear dynamics

Generally, nonlinear ODEs do not have closed form solutions!

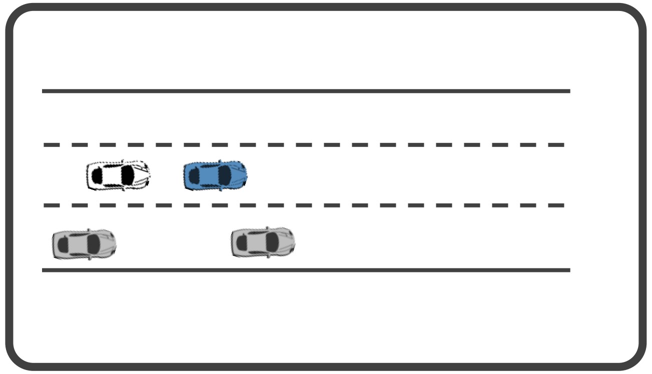
Dubin's car model

$\dot{v} = a$	Speed
$\frac{ds_x}{dt} = v\cos(\psi)$	Horizontal position
$\frac{ds_y}{dt} = v\sin(\psi)$	Vertical position
$\frac{d\delta}{dt} = v_\delta$	Steering angle
$\frac{d\psi}{dt} = \frac{v}{l}\tan(\delta)$	Heading angle

Physical plant

$\frac{dx}{dt} = f(x,u)$ $x[t+1] = f(x[t],u[t])$	System dynamics
$x = [v, s_x, s_y, \delta, \psi]$	State variables
$u = [a, v_\delta]$	Control inputs

Nonlinear *hybrid* dynamics



Physical plant

$\frac{dx}{dt} = f(x,u)$

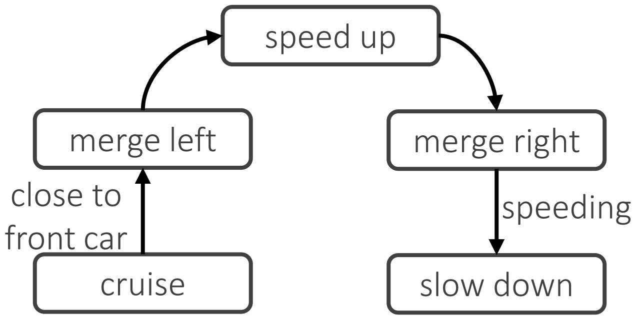
System dynamics

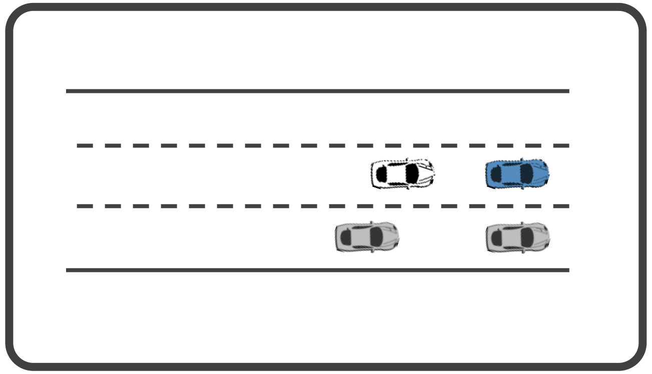
$x[t+1] = f(x[t], u[t])$

$x = [v, s_x, s_y, \delta, \psi]$ State variables

$u = [a, v_\delta]$ Control inputs

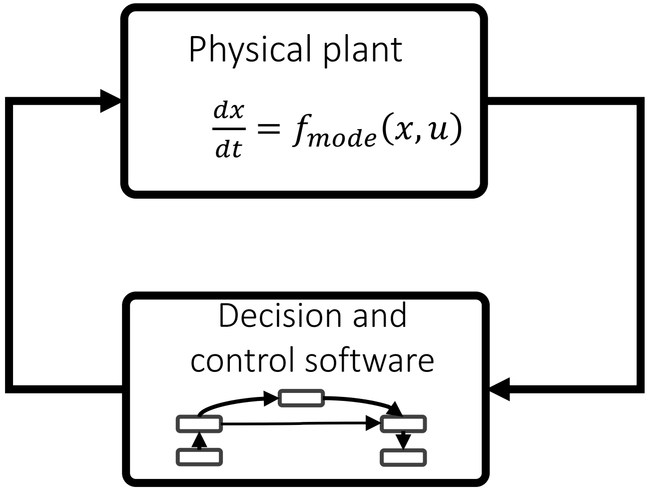
Decision and control software

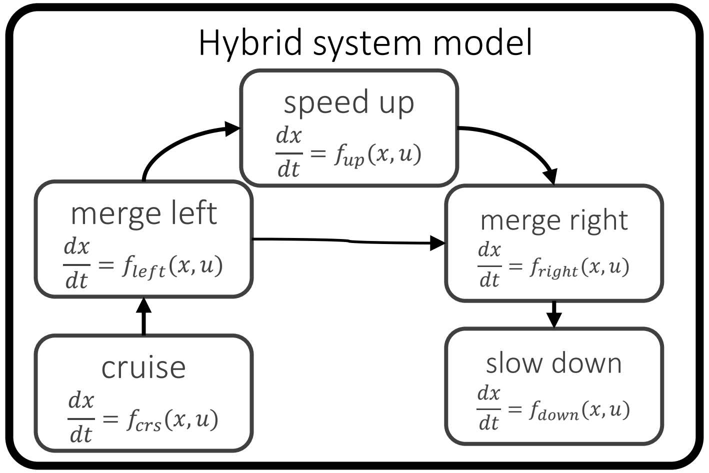




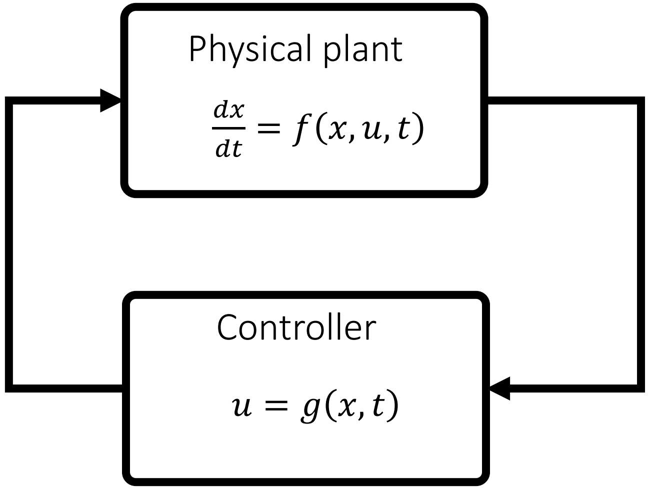
Nonlinear *hybrid* dynamics

Interaction between computation and physics can lead to unexpected behaviors





Simplified view of a plant and a controller



$$\frac{dx}{dt} = f(x, g(x), t)$$

$$\dot{x} = f(x, g(x), t)$$

Model?

A set of mathematical relationships among the system variables

Dynamical Systems Model

Describe behavior in terms of instantaneous laws

$$\frac{dx(t)}{dt} = f(x(t), u(t), t)$$

$t \in \mathbb{R}, x(t) \in \mathbb{R}^n, u(t) \in \mathbb{R}^m$

$f: \mathbb{R}^n \times \mathbb{R}^m \times \mathbb{R} \to \mathbb{R}^n$ dynamic function

Example: Pendulum

Pendulum equation

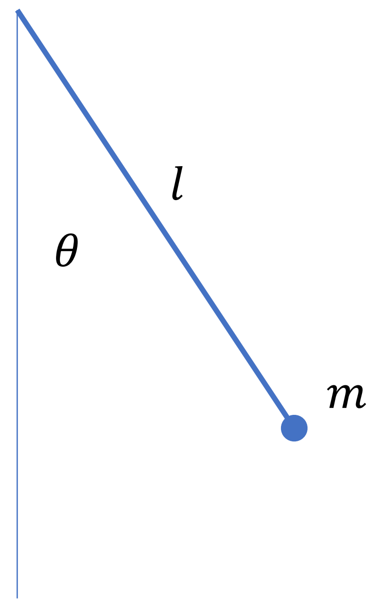
$x_1 = \theta \quad x_2 = \dot{\theta}$

$x_2 = \dot{x}_1$

$\dot{x}_2 = -\frac{g}{l}\sin(x_1) - \frac{k}{m}x_2$

$\begin{bmatrix}\dot{x}_2 \\ \dot{x}_1\end{bmatrix} = \begin{bmatrix}-\frac{g}{l}\sin(x_1) - \frac{k}{m}x_2 \\ x_2\end{bmatrix}$

k: friction coefficient

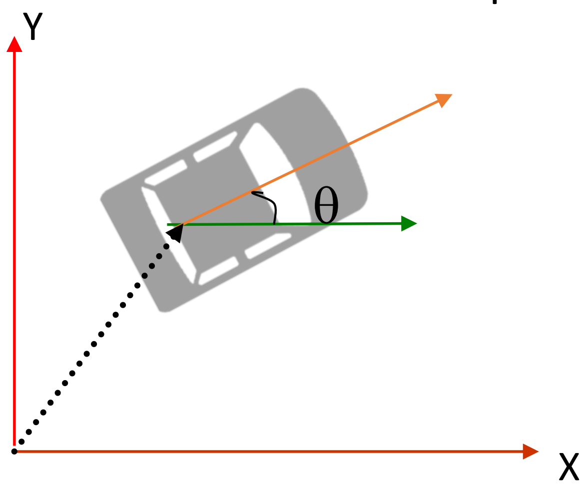


What is described?

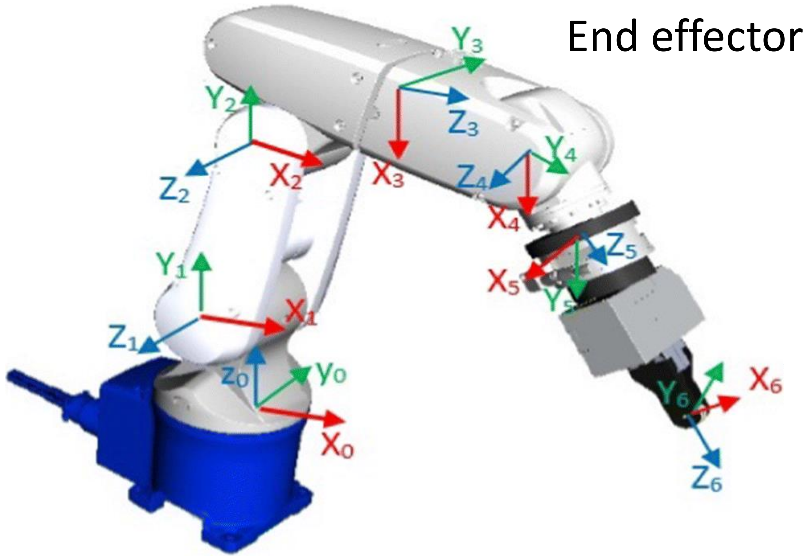
-> Center of mass movement relative to the origin

Coordinate system

- Configuration (pose) of robot can be described by position and orientation.

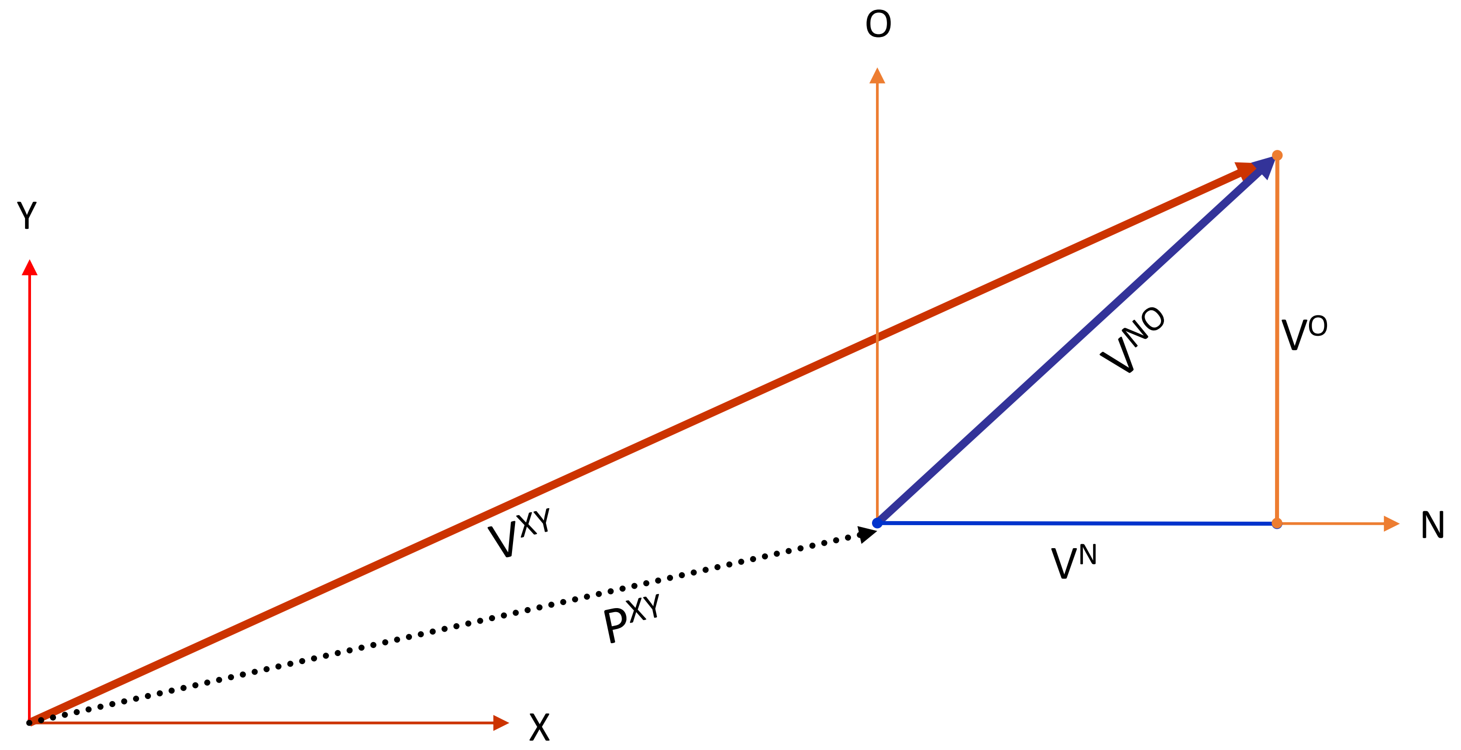


$$\text{Car pose} = \begin{bmatrix} x \\ y \\ \theta \end{bmatrix}$$



$$\text{End effector pose} = \begin{bmatrix} x \\ y \\ z \\ \theta_x \\ \theta_y \\ \theta_z \end{bmatrix}$$

Translation along the X-Axis and Y-Axis

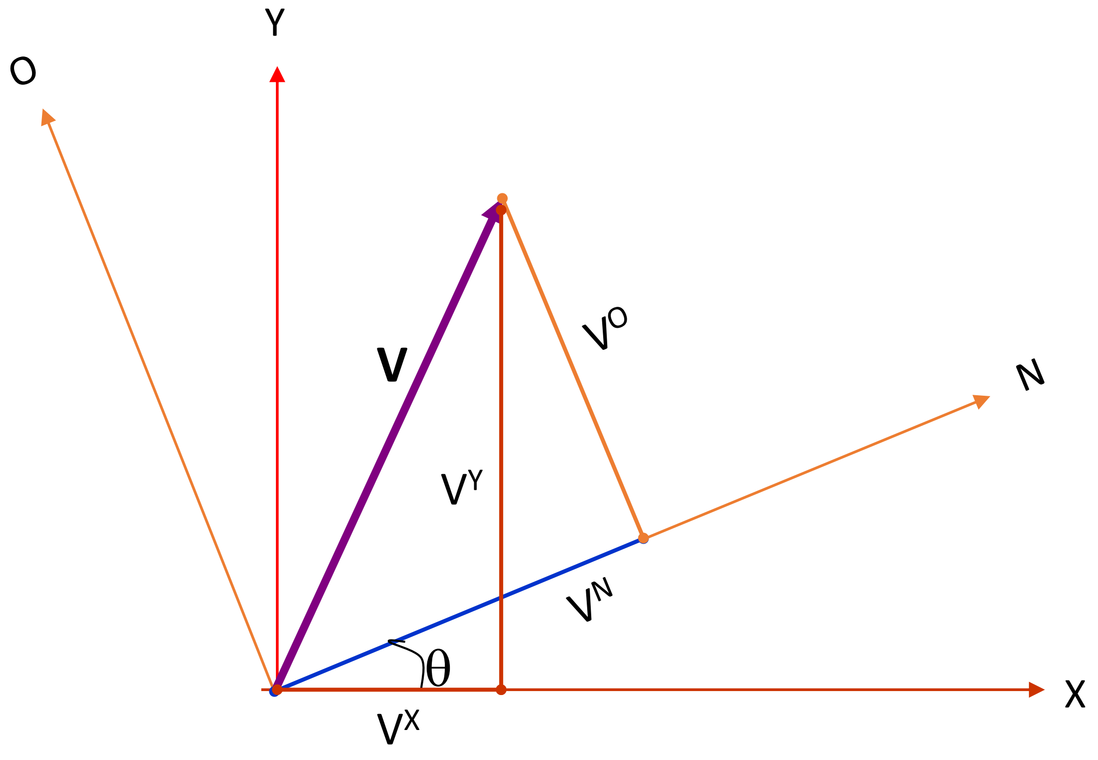


$$\bar{P}^{XY} = \begin{bmatrix} P_x \\ P_Y \end{bmatrix}$$

$$\bar{V}^{NO} = \begin{bmatrix} V^N \\ V^O \end{bmatrix}$$

$$\bar{V}^{XY} = \bar{P}^{XY} + \bar{V}^{NO} = \begin{bmatrix} P_X + V^N \\ P_Y + V^O \end{bmatrix}$$

Rotation (Z-Axis)



θ = Angle of rotation between the XY and NO coordinate axis

$$\bar{V}^{XY} = \begin{bmatrix} V^X \\ V^Y \end{bmatrix} \qquad \bar{V}^{NO} = \begin{bmatrix} V^N \\ V^O \end{bmatrix}$$

$$\bar{V}^{XY} = \begin{bmatrix} V^X \\ V^Y \end{bmatrix} = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix} \begin{bmatrix} V^N \\ V^O \end{bmatrix}$$

- Rotation matrix

$$R(\theta) = \begin{bmatrix} cos\theta & -sin\theta \\ sin\theta & cos\theta \end{bmatrix}$$

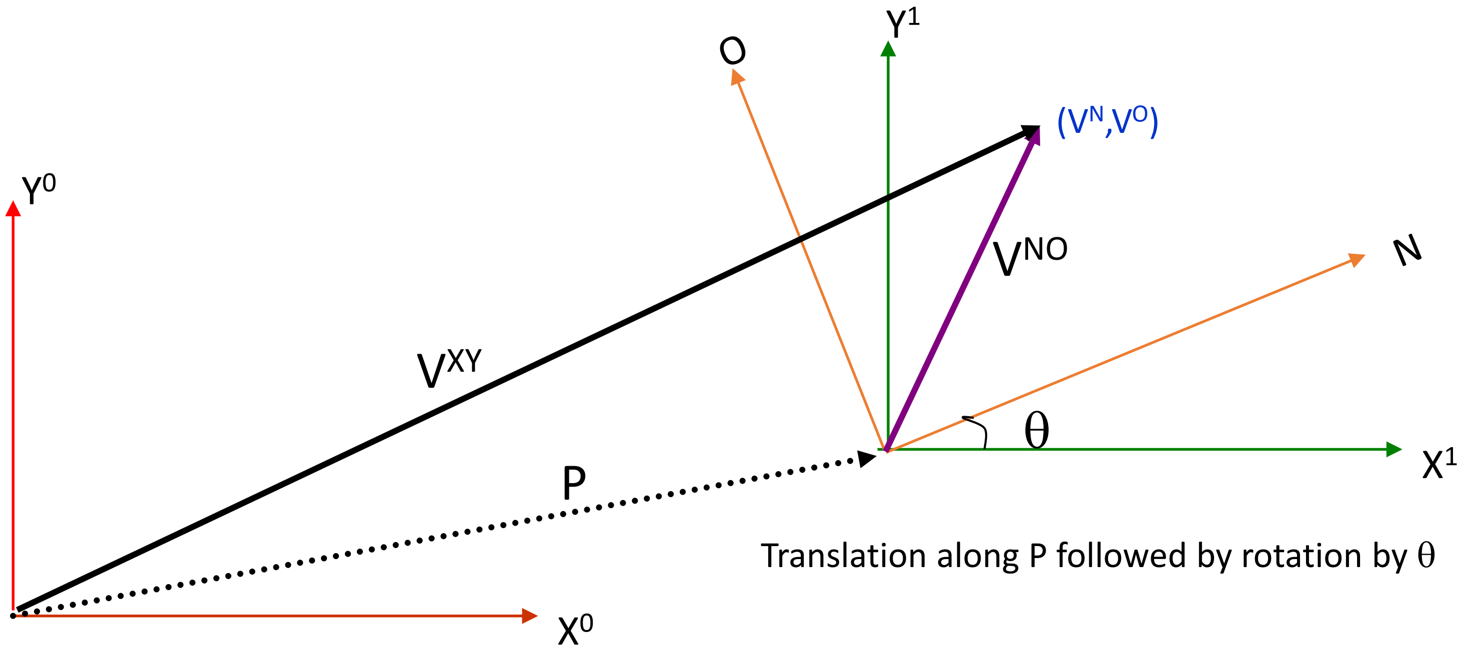
  - An orthogonal (orthonormal) matrix
    - Each column is a unit length vector
    - Each column is orthogonal to all other columns
  - The inverse is the same as the transpose

$$R(\theta)^{-1} = R(\theta)^T$$

  - The determinant is 1
  - Special Orthogonal group of dimension 2

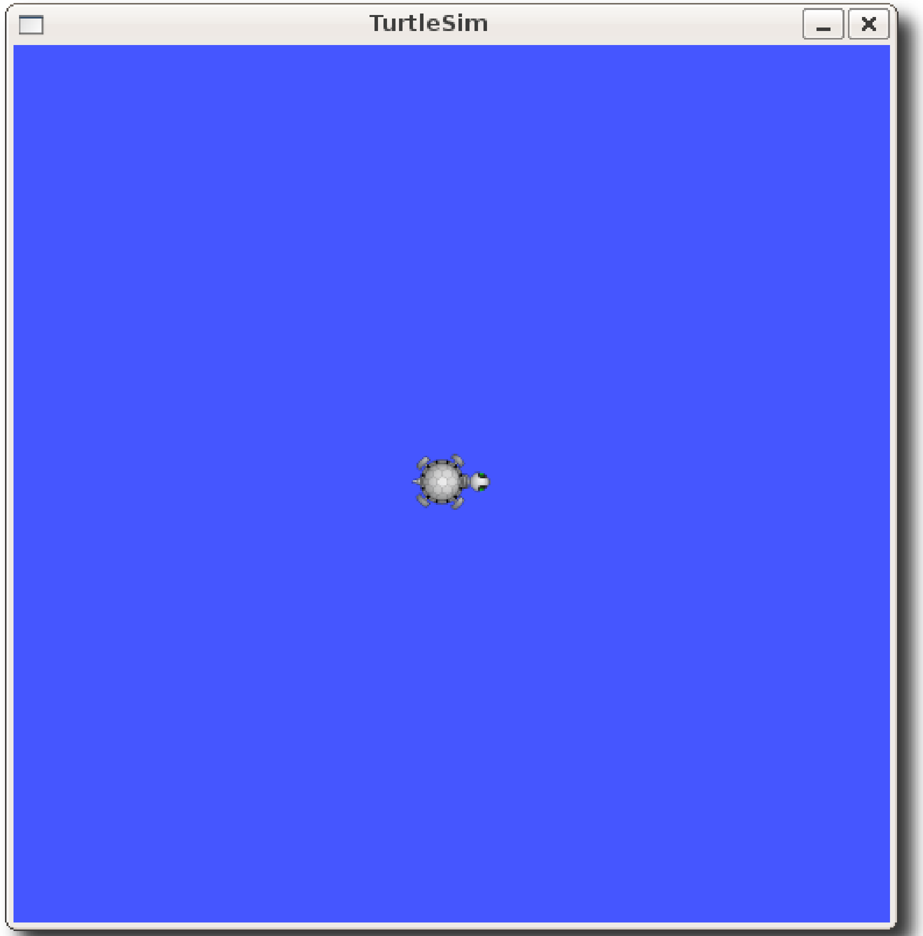
$$R(\theta) \in SO\{2\}$$

Transformation



$$V^{XY} = \begin{bmatrix} V^X \\ V^Y \end{bmatrix} = \begin{bmatrix} P_x \\ P_y \end{bmatrix} + \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix} \begin{bmatrix} V^N \\ V^O \end{bmatrix}$$

TurtleSim

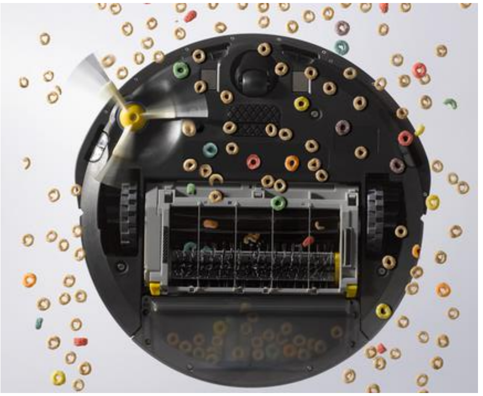


- Turtle simulator
- simple way to learn the basics of ROS
- a ROS package, part of the ROS installation

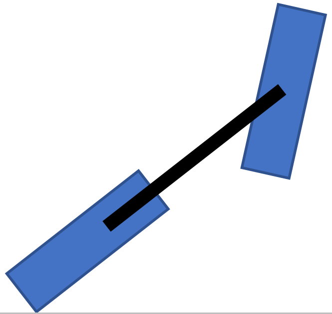
- Explore the nodes, topics, messages, and services (roscore, rosnode, and rostopic commands)

Is ROS important?

Locomotion of Wheeled System

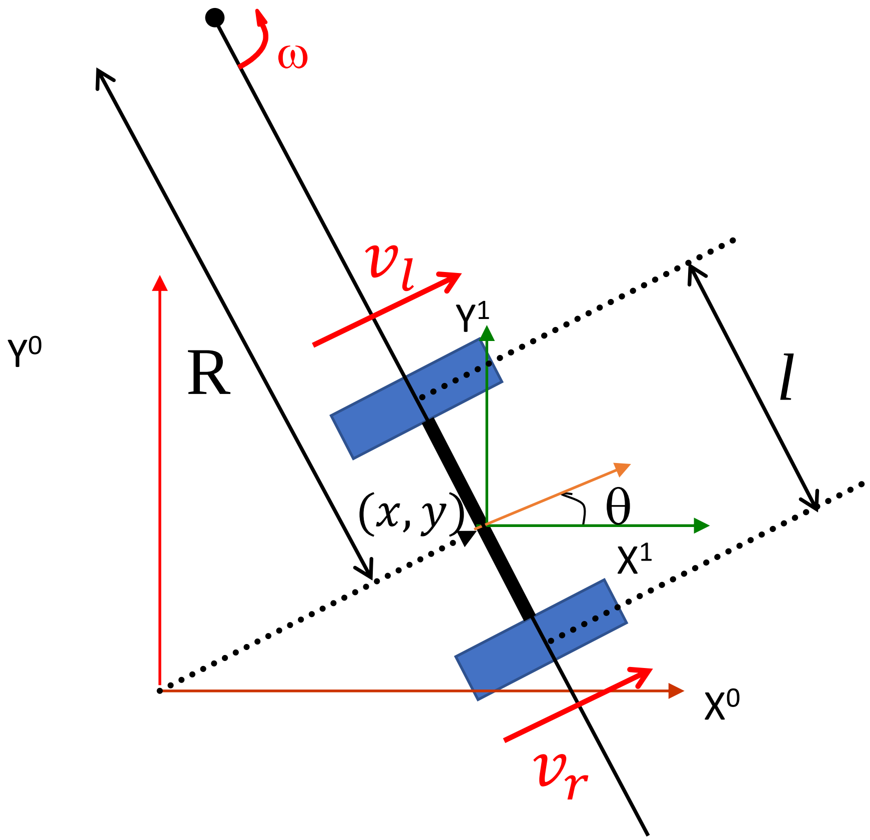


Differential Drive Model



Rear Wheel Model

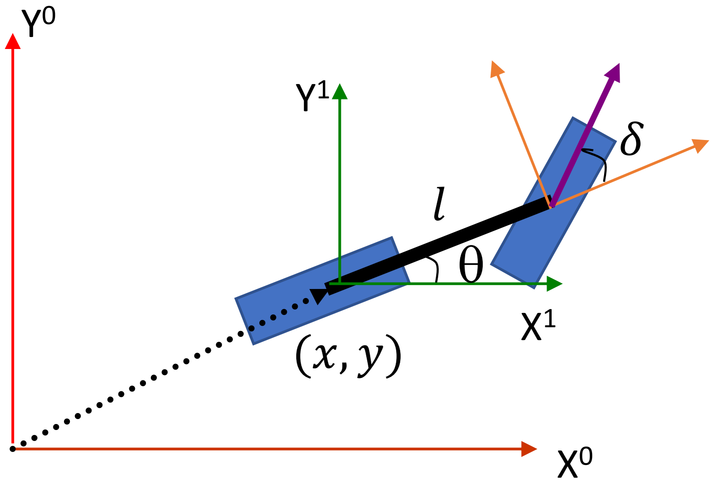
Differential Drive Model



Instantaneous Center of Curvature
$= [x - R\sin\theta, y + R\cos\theta] = [ICC_x, ICC_y]$

$$\omega(R + l/2) = v_r$$
$$\omega(R - l/2) = v_l$$
$$R = \frac{l}{2}\frac{(v_r + v_l)}{(v_r - v_l)}$$
$$\omega = \frac{v_r - v_l}{l}$$

Rear Wheel Model (Dubin's model)



Car length = l

Car (real wheel) pose = $\begin{bmatrix} x \\ y \\ \theta \end{bmatrix}$

Car speed = v

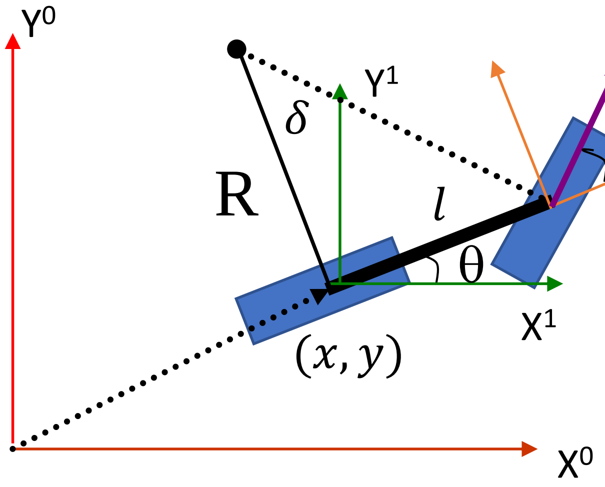
Car (front wheel) steering angle = δ

$$\dot{x} = v\ cos\theta$$

$$\dot{y} = v\ sin\theta$$

$$\dot{\theta} = \frac{v}{l}\ tan\delta$$

Rear Wheel Model (Dubin's model)



Car length = l

Car (real wheel) pose = $\begin{bmatrix} x \\ y \\ \theta \end{bmatrix}$

Car speed = v

Car (front wheel) steering angle = δ

$$\dot{x} = v\ cos\theta$$

$$\dot{y} = v\ sin\theta$$

$$\dot{\theta} = \frac{v}{l}\ tan\delta$$

$$Rd\theta = dp_r$$

$$R\frac{d\theta}{dt} = \frac{dp_r}{dt} = v$$

$$tan\delta = \frac{l}{R} \Rightarrow R = \frac{l}{tan\delta}$$