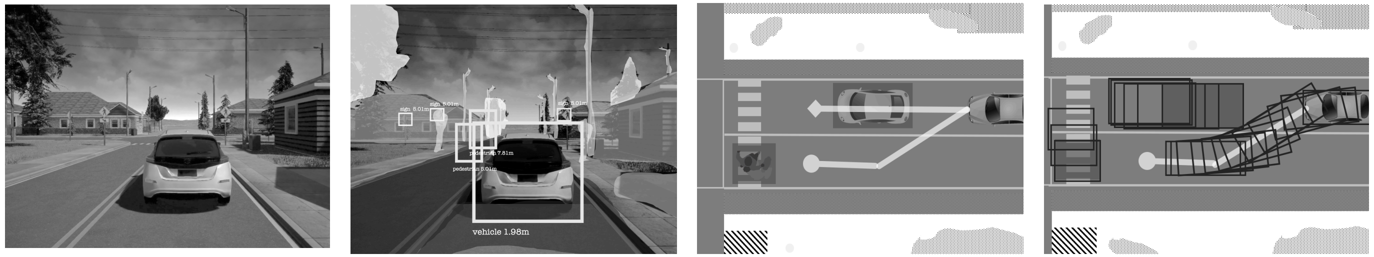
ECE484 Principles of Safe Autonomy Lecture 7

Modeling and Control

Sayan Mitra

GEM platform

Autonomy pipeline



Sensing

Physics-based models of camera, LIDAR, RADAR, GPS, etc.

Perception

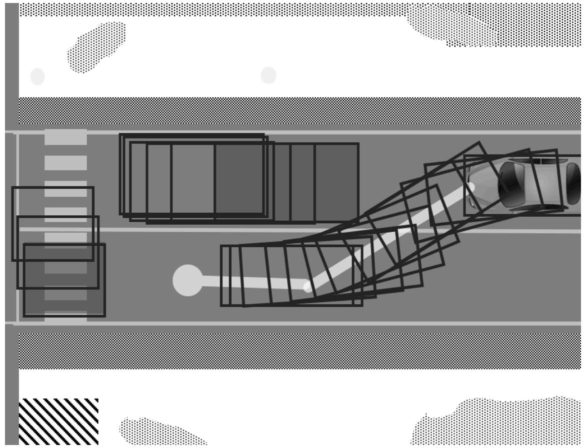
Programs for object detection, lane tracking, scene understanding, etc.

Decisions and planning

Programs and multi-agent models of pedestrians, cars, etc.

Control

Dynamical models of engine, powertrain, steering, tires, etc.



Control

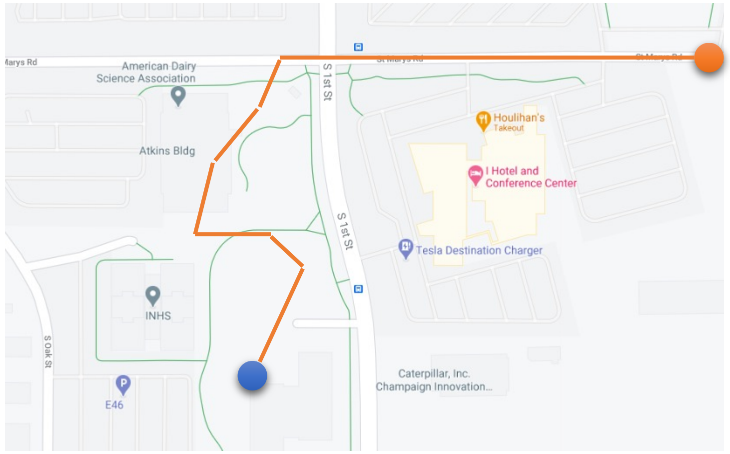
Dynamical models of engine, powertrain, steering, tires, etc.

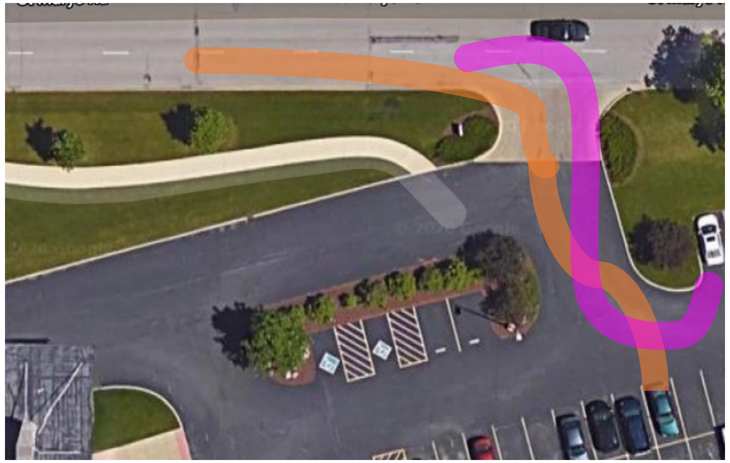
Outline

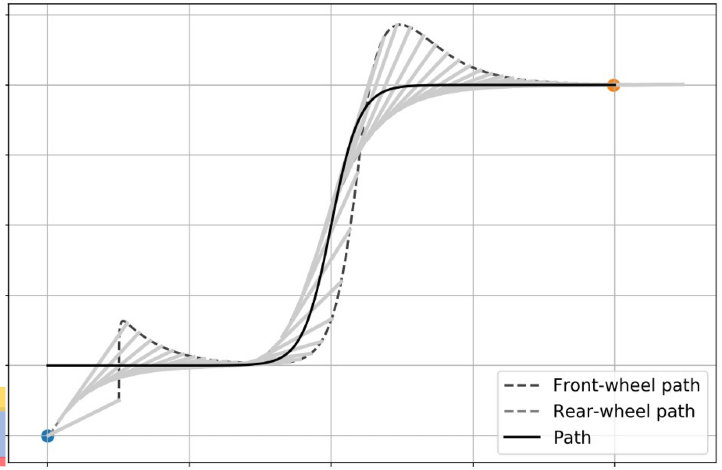
- Modeling the control problem
  - Differential Equations; solutions and their properties
- Control design
  - Open loop vs closed loop
  - PID
  - State feedback
  - MPC (brief)
- Requirements
  - Stability
  - Lyapunov theory and its relation to invariance

Typical planning and control modules

- Global navigation and planner
  - Find paths from source to destination with static obstacles
  - Algorithms: Graph search, Dijkstra, Sampling-based planning
  - Time scale: Minutes
  - Output: reference center line, does not consider vehicle dynamics
- Local planner
  - Dynamically feasible trajectory generation
  - Dynamic planning w.r.t. obstacles
  - Time scales: 10 Hz
- Controller
  - Waypoint follower using steering, throttle
  - Algorithms: PID control, MPC, Lyapunov-based controller
  - Lateral/longitudinal control
  - Time scale: 100 Hz







What is control

Control theory is the *art* of making *things* do what *you want* them to do

art: tuning parameters

things: Differential equation models

what you want: tracking error or stability

Open look control

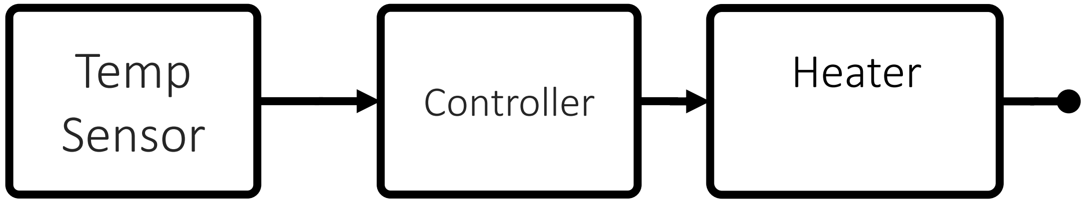
System: Sensor, control logic, heater

Control logic: Check every 30 mins

If temperature $\theta_s \leq 70$ then run heater for the next 30 mins;

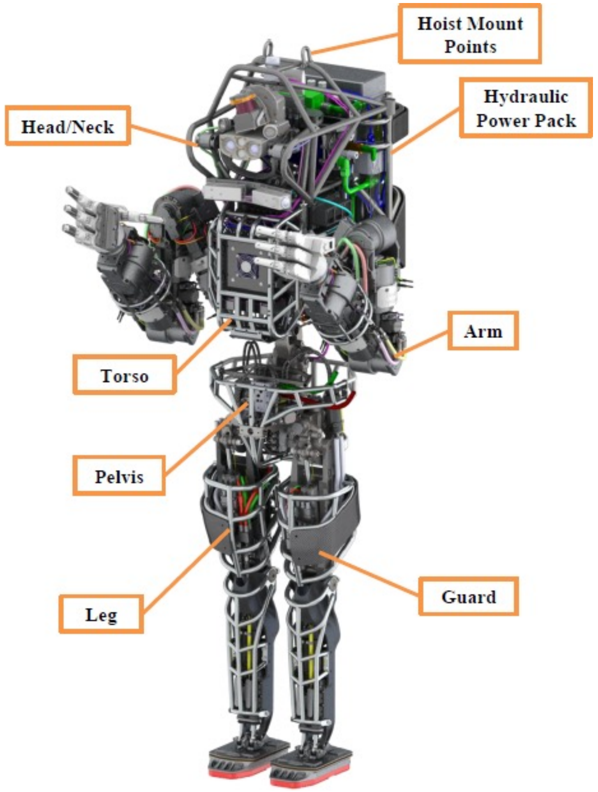
if $\theta_s \geq 75$ then turn off heater for the next 30 mins

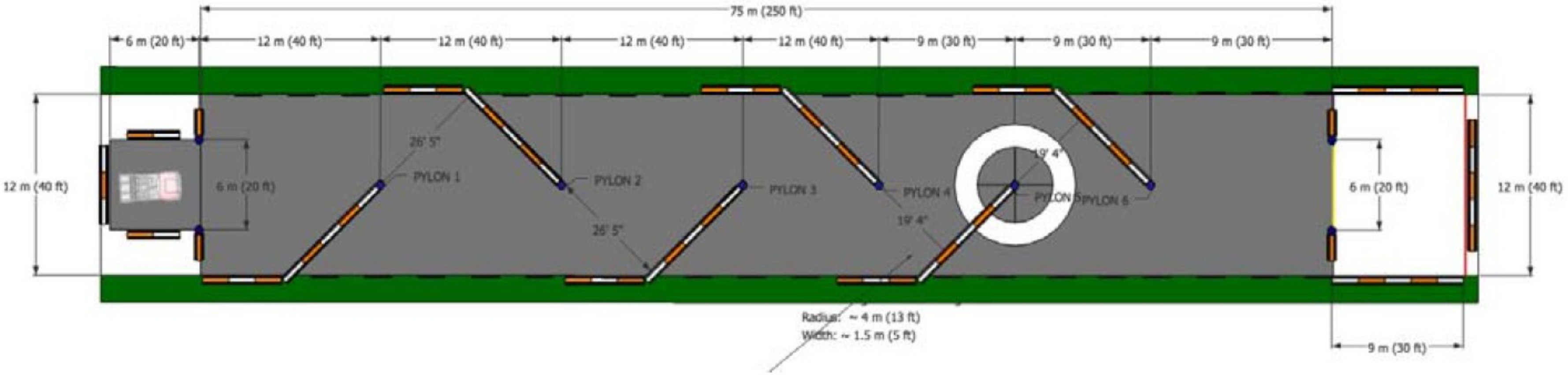
Open loop: output of the system is not used by the controller



Complex control tasks: DARPA Robotics Challenge

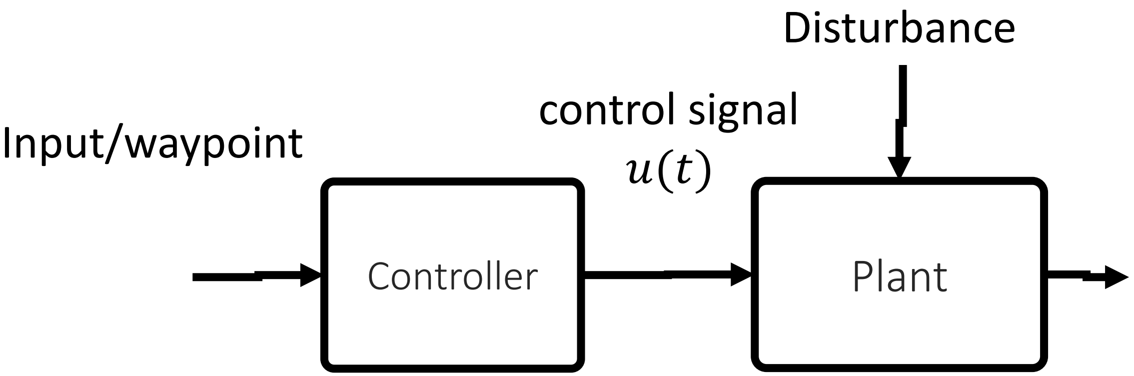
- 4 points task
  - Robot drives the vehicle through the course (1)
  - Robot gets out of the vehicle and travels dismounted out of the end zone (2)
  - Bonus point (1)



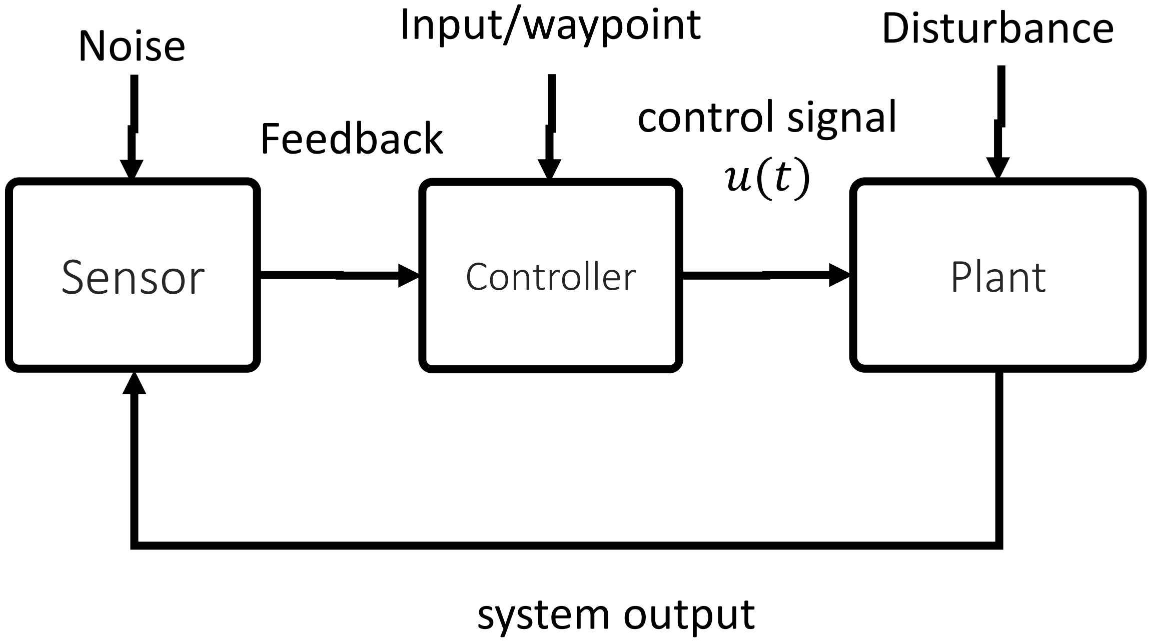



POLARIS
2

Open & Closed loop control



$$\dot{x}(t) = f\big(x(t), u(t)\big)$$
$$y(t) = h(x(t))$$
$$u(t) = g(y_d(t))$$

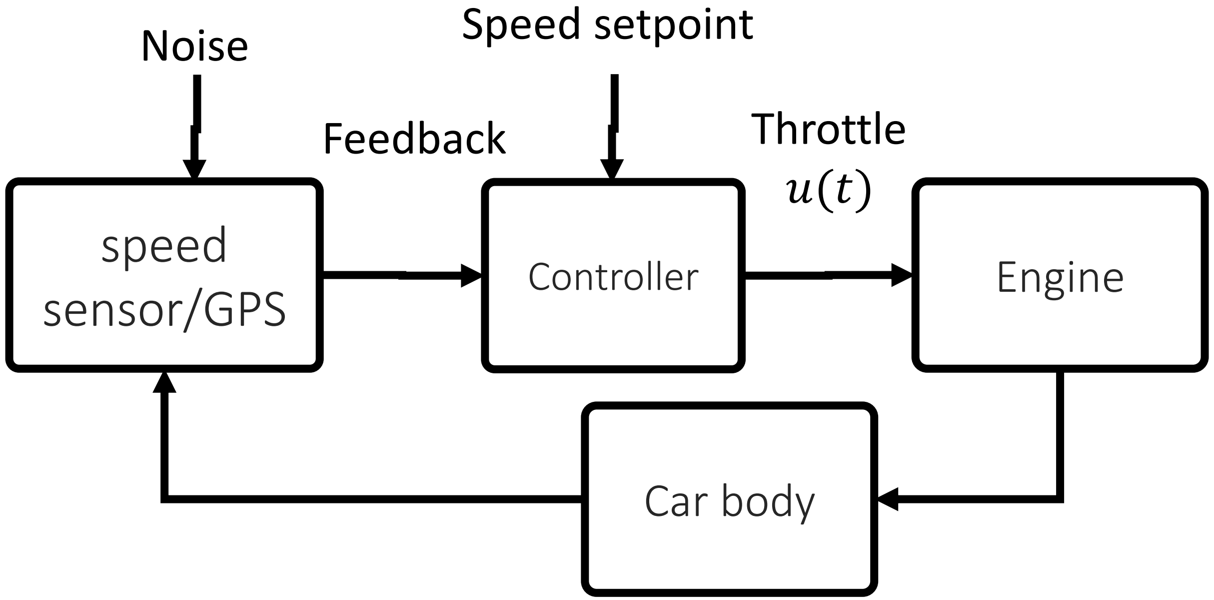


$$\dot{x}(t) = f\big(x(t), u(t)\big)$$
$$y(t) = s\big(x(t)\big)$$
$$u(t) = g(y(t), y_d(t))$$

Cruise control

$$\dot{x}(t) = f\big(x(t), u(t)\big)$$
$$y(t) = s\big(x(t)\big)$$
$$u(t) = g(y(t), y_d(t))$$

Control design is the problem of figuring out g given certain requirements on $y(t)$



Modeling control systems

Behaviors of physical processes are described in terms of instantaneous laws

Common notation: $\frac{dx(t)}{dt} = f(x(t), u(t), t) \; - (1)$,

where time $t \in \mathbb{R}$; state $x(t) \in \mathbb{R}^n$; $input\ \ u(t) \in \mathbb{R}^m$; $f: \mathbb{R}^n \times \mathbb{R}^m \times \mathbb{R} \rightarrow \mathbb{R}^n$

Example. $\frac{dx(t)}{dt} = v(t)$; $\frac{dv(t)}{dt} = -g$

Initial value problem: Given system (1) and initial state $x_0 \in \mathbb{R}^n, t_0 \in \mathbb{R}$, and input u: $\mathbb{R} \rightarrow \mathbb{R}^m$, find a state trajectory or *solution* of (1).

Example 1: Pendulum

Pendulum equation

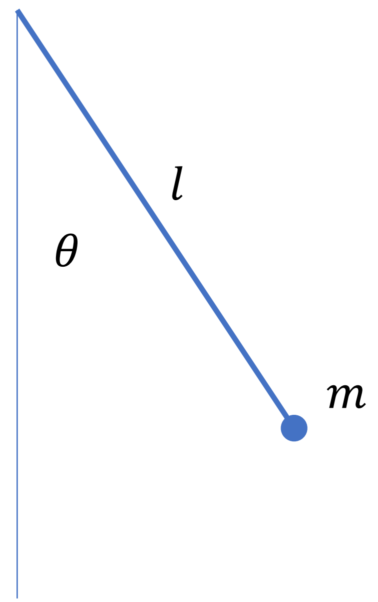
$x_1 = \theta \quad x_2 = \dot{\theta}$

$x_2 = \dot{x}_1$

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

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

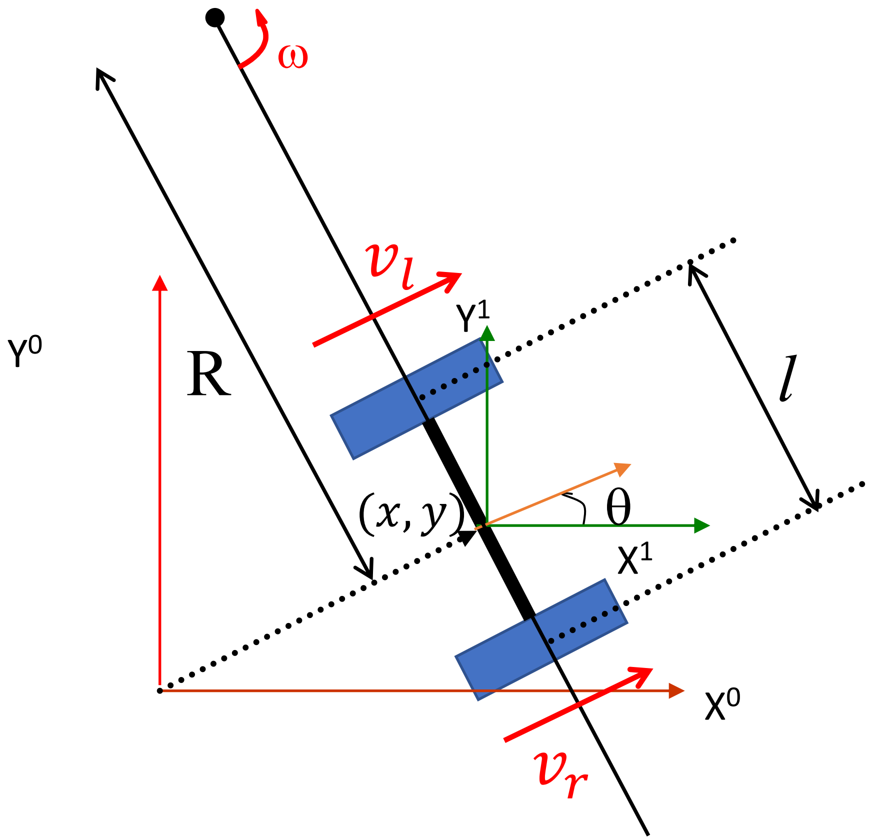
k: friction coefficient



What is described?

-> Center of mass movement relative to the origin

Example 2. Differential Drive Model



Instantaneous Center of Curvature

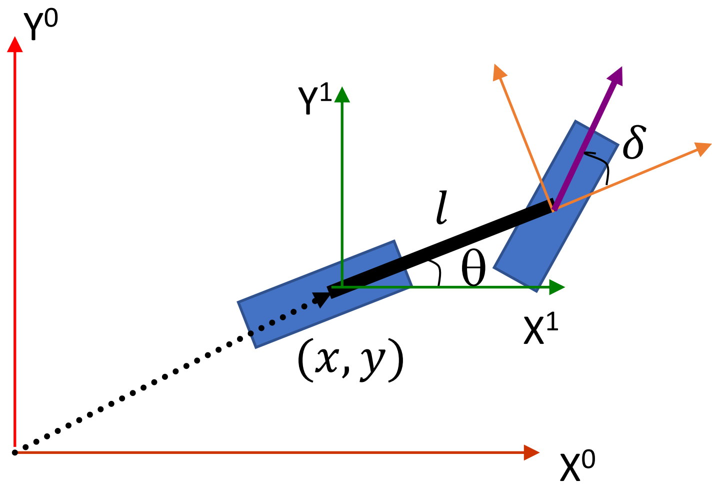
$= [x - R\sin\theta, y + R\cos\theta] = [ICC_x, ICC_y]$

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

Example 3: Simple vehicle model: Dubin's car

Key assumptions

- Front and rear wheel in the plane in a stationary coordinate system
- Steering input, front wheel steering angle δ
- No slip: wheels move only in the direction of the plane they reside in

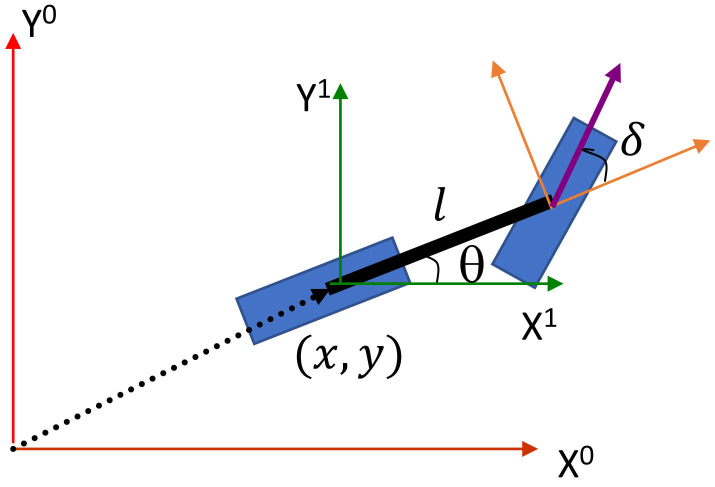


Zeroing out the accelerations perpendicular to the plane in which the wheels reside, we get the equations in the next slide

Modeling one wheel is enough

Reference: Paden, Brian, Michal Cap, Sze Zheng Yong, Dmitry S. Yershov, and Emilio Frazzoli. 2016. A survey of motion planning and control techniques for self-driving urban vehicles. IEEE Transactions on Intelligent Vehicles 1 (1): 33–55.

Rear Wheel Model (Dubin's model)



Car length = l

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

Car speed = v

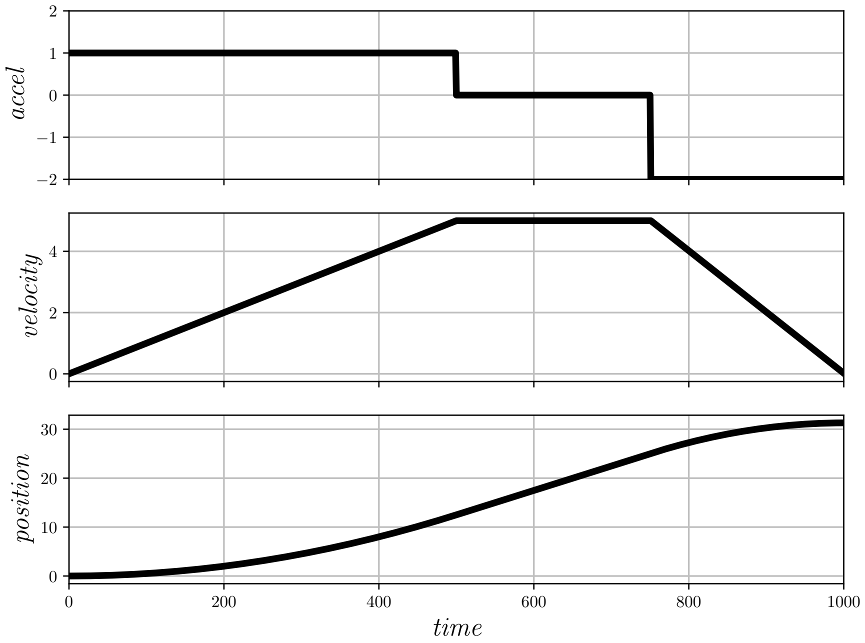
Car (front wheel) steering angle = δ

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

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

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

Notions of solution



What is a solution? Many different notions.

Definition 1. (First attempt) Given x_0 and u, $\xi: \mathbb{R} \to \mathbb{R}^n$ is a solution o trajectory iff

(1) $\xi(t_0) = x_0$ and

(2) $\frac{d}{dt}\xi(t) = f(\xi(t), u(t), t)), \forall t \in \mathbb{R}$.

Mathematically OK, but too restrictive for autonomous systems.

Assumes that ξ is not only continuous, but also differentiable. This disallows $u(t)$ to be discontinuous, which is often required for optimal control.

Is PC input $u(t)$ adequate for guaranteeing existence of solutions?

Example. $\dot{x}(t) = -sgn(x(t)); x_0 = c; t_0 = 0; c > 0$
Solution: $\xi(t) = c - t$ for $t \leq c$; check $\dot{\xi}(t) = -1 = -\text{sgn}(\xi(t))$
Problem: Solution undefined at $t = c$, f discontinuous in x

Example. $\dot{x}(t) = x^2; x_0 = c; t_0 = 0; c > 0$
Solution: $\xi(t) = \frac{c}{1-tc}$ works for $t < 1/c$; check $\dot{\xi}$
Problem: As $t \to \frac{1}{c}$ then $x(t) \to \infty$; p grows too fast

No, we need stronger conditions on smoothness of $f(.)$

Lipschitz continuity

A function $f: \mathbb{R}^n \rightarrow \mathbb{R}$ is Lipschitz continuous if there exist $L > 0$ such that for any pair $x, x' \in \mathbb{R}^n$, $\left|\left|f(x) - f(x')\right|\right| \leq L\left|\left|x - x'\right|\right|$

Examples: $6x + 4$; $|x|$; all differentiable functions with bounded derivatives

Are Lipschitz continuous functions closed under addition, multiplication?

Non-examples: $\sqrt{x}$; x^2 (locally Lipschitz)

Dynamical Systems Model

Describe behavior in terms of instantaneous laws

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

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

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

Theorem. If $f(x(t), u(t))$ is Lipschitz continuous in the first argument and $u(t)$ is piece-wise continuous then (1) has unique solutions.

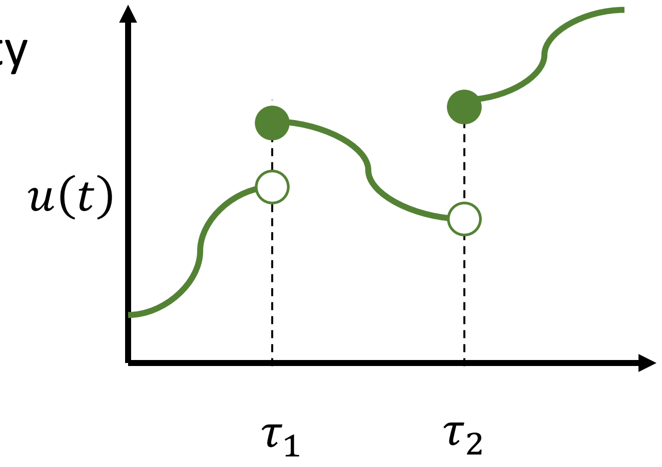
Modified notion of solution*

Definition. $u(\cdot)$ is a *piece-wise continuous* with set of discontinuity points $D \subseteq \mathbb{R}^m$ if

(1) $\forall \tau \in D, \lim_{t \to \tau^+} u(t) < \infty; \lim_{t \to \tau^-} u(t) < \infty$

(2) Continuous from right $\lim_{t \to \tau^+} u(t) = u(t)$

(3) $\forall\, t_0 < t_1$, $[t_0, t_1] \cap D$ is finite



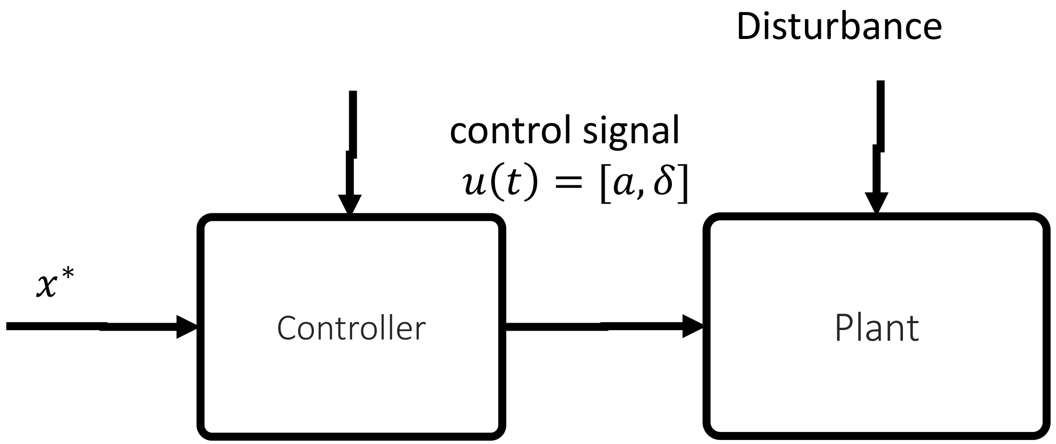
$PC([t_0, t_1], \mathbb{R}^m)$ is the set of all piece-wise continuous functions over the domain $[t_0, t_1]$

Definition 2. Given x_0 and u, $\xi: \mathbb{R} \to \mathbb{R}^n$ is a solution or trajectory iff (1) $\xi(t_0) = x_0$ and (2) $\frac{d}{dt}\xi(t) = f(\xi(t), u(t), t), \forall t \in \mathbb{R} \backslash \mathrm{D}$.

Since $u(t)$ is piece-wise continuous, so is f in the second argument

Control design

Open loop control



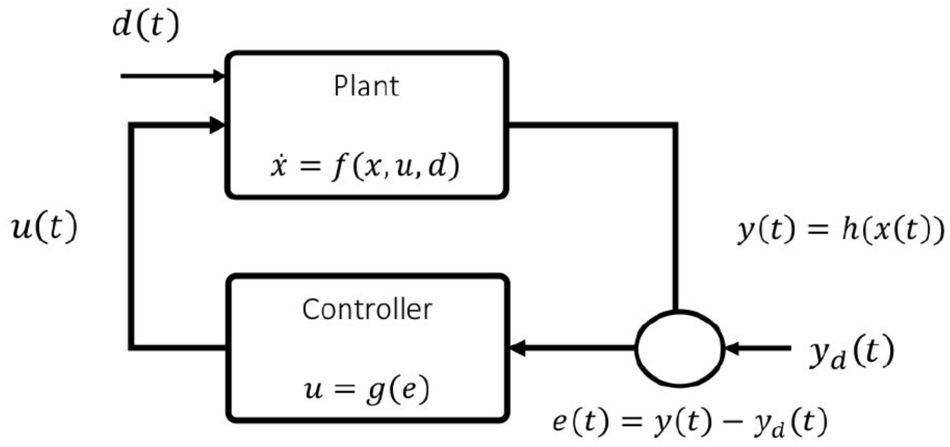
PID control strategy

- Error measured as difference between desired output and measured output
  - $e(t) = y_d(t) - y(t)$
- Control input defined as terms proportional to error, integral of error, and derivative of error
- $u(t) = K_P e(t) + K_I \int_0^t e(\tau)d\tau + K_D \frac{de(t)}{dt}$

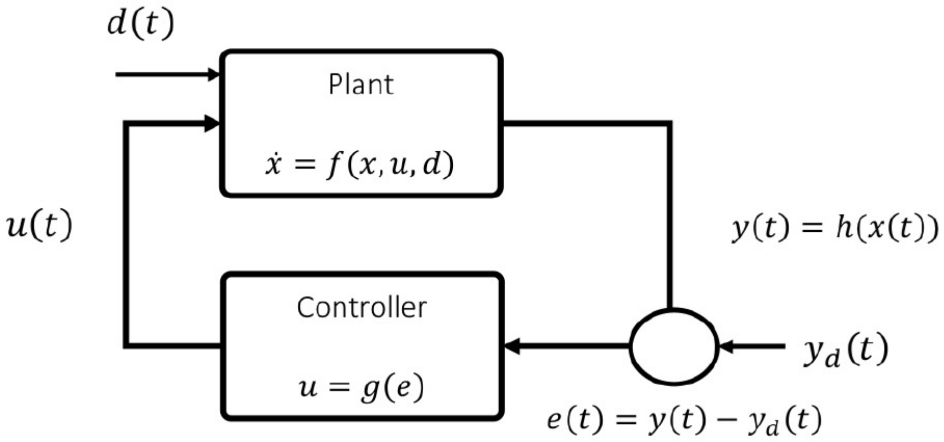
  $= K_P \left(e(t) + \frac{1}{T_I} \int_0^t e(\tau)d\tau + T_D \frac{de(t)}{dt} \right)$
- It is necessary to tune these gains of the PID controller
  - If the gain is too high the system may become unstable
  - If the gain is too low the system may not respond
- PD control: $K_I = 0$ or $T_I = \infty$
- PI control: $K_D = 0 = T_D$
- P control: $K_I = 0$ and $K_D = 0$
  - Steady state may not be 0

A simple P-controller example

- $\dot{y}(t) = u(t) + d(t)$
- Using proportional (P) controller
- $u(t) = -K_P e(t) = -K_P(y(t) - y_d(t))$
- $\dot{y}(t) = -K_P y(t) + K_P y_d(t) + d(t)$
- Consider constant setpoint y_0 and disturbance d_{ss}
- $\dot{y}(t) = -K_P y(t) + K_P y_0 + d_{ss}$
- What is the steady state output?
  - Set $-K_P y(t) + K_P y_0 + d_{ss} = 0$
  - $y(t) = y_{ss} = \frac{d_{ss}}{K_P} + y_0$



A simple P-controller example



- $\dot{y}(t) = u(t) + d(t)$
- Consider constant setpoint y_0 and disturbance d_{ss}
- $\dot{y}(t) = -K_P y(t) + K_P y_0 + d_{ss}$
- Steady state output $y(t) = y_{ss} = \frac{d_{ss}}{K_P} + y_0$
- Transient behavior
  - $y(t) = y(0)e^{-t/T} + y_{ss}\left(1 - e^{-\frac{t}{T}}\right), T = 1/K_P$
- To make steady state error small we can increase K_P at the expense of longer transients

Summary

- ODE language for control systems
  - Solutions, Lipschitz continuity, equilibria, steady state, transient
- Control design
  - Open loop vs closed loop
  - PID design