2. Autonomous Driving: Speed Tracking

金力 Li Jin
li.jin@sjtu.edu.cn
上海交通大学密西根学院
Shanghai Jiao Tong University UM Joint Institute

Recap

- Keywords
- Applications: Smart cities/cars/grid/living…
- Tools: learning/control/optimization

Question for you

Why in the first place are we interested in autonomous driving?

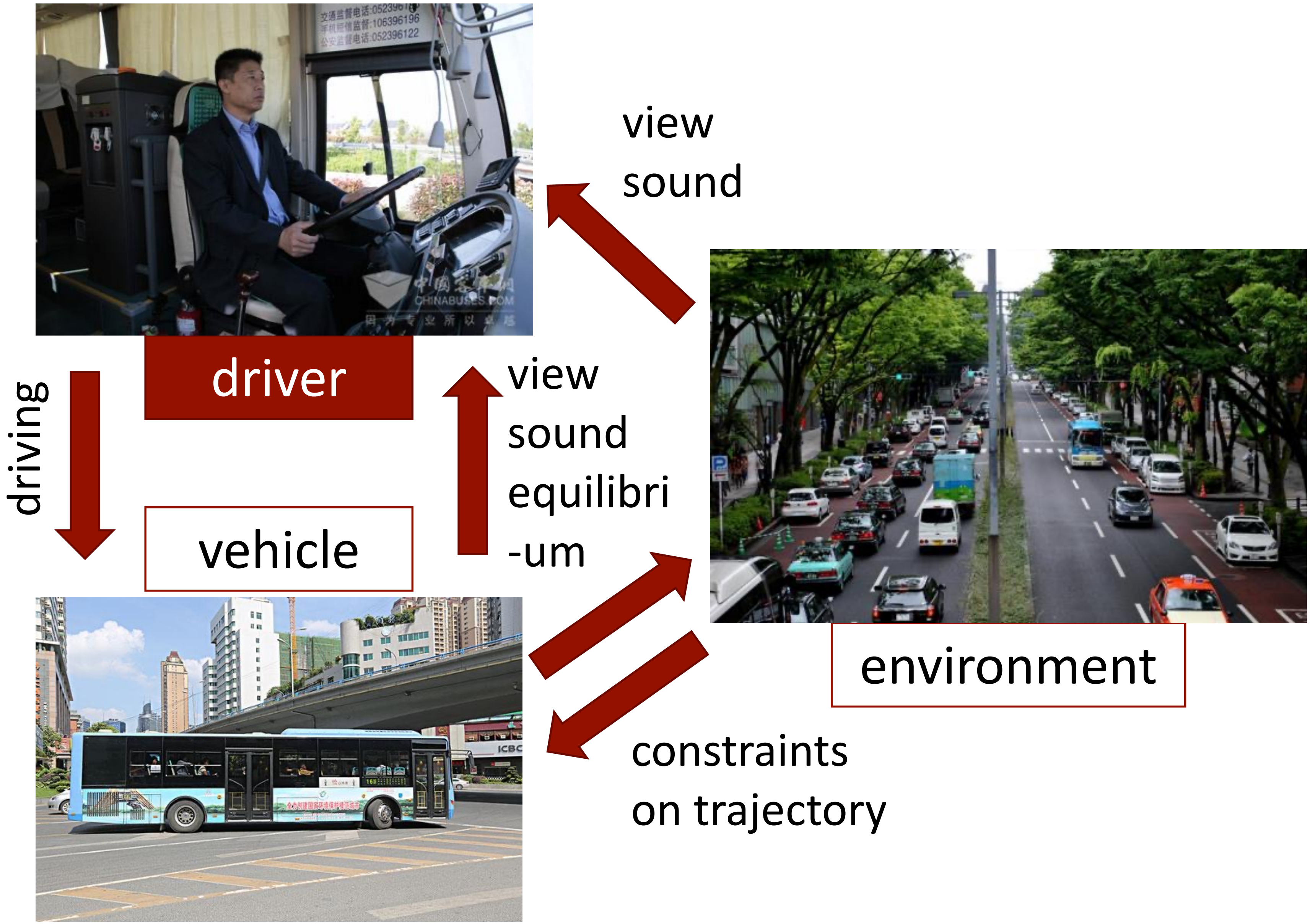
Outline

- Concept of autonomous driving
  - How human drive
  - How a computer drives
  - Decision-making architecture
- Speed tracking
  - State, dynamics, action, policy
  - Performance of policy
- 1D dynamical system & control
  - Modeling
  - Objective
  - Theory*

How human drive

What does a driver do when he/she drives?

What does a driver do when he/she drives?



What does a driver do when he/she drives?

- Perceive the environment
  - Navigate and plan route
  - Identify road lanes & markings
  - Recognize traffic signs & signals
  - Watch for pedestrians & bikes
  - Observe movement of neighboring vehicles

- Take actions
  - Select route
  - Keep or change lanes
  - Rotate steering wheel
  - Push/release throttle/brake/*clutch*
  - Change gear
  - Signaling

How do human make decisions?

Human drivers have objectives:

- Arrive at destination as soon as possible
- Ensure no collision with other vehicles/bikes/pedestrians/obstacles…
- Drive smoothly to ensure comfort
- Avoid traffic rule violation
- Take as few actions as possible

How a computer drives

What does a computer do when it drives?

Perception

- Navigation tool (e.g. Amap 高德地图)
- Camera
- Radar
- LiDAR (Light Detection and Ranging)
- Infrared detector
- Speedometer
- Engine speed sensor
- Vehicle-to-vehicle communications
- Vehicle-to-infrastructure communications

Actuation

- Itinerary planning
- Route selection
- Maneuver planning: lane keeping, lane changing, overtaking
- Longitudinal control: throttle/brake position or engine torque
- Lateral control: steering wheel position or tire position

Decision-making

- What is a good time for departure?
- What is a good route?
- What is a good lane to travel on?
- What is a good driving style? As fast as possible? Consume as little fuel as possible?
- What is a good way to accelerate/decelerate?
- What is a good way to make a turn?
- What is a good parking location?

Decision making architecture

DRIVER DECISIONS AND IVHS FUNCTIONS

Phase	Driver decision	IVHS goal	IVHS task	Strategy
Pre-trip	Trip generation, modal choice, etc	More efficient resource utilization	Demand shift	I, A, P
In-trip	Route choice	Reduce travel time	Route guidance and flow control	I, A, C
	Path planning	Smooth traffic	Congestion control	I, A, C
	Maneuver	Increase safety, flow	Vehicle coordination	I, C
	Regulation	Increase safety, flow	Proper spacing, steering, etc.	I, C
Post-trip	Parking, etc	Add value	Efficient use	I, A, C, P

A = Advice, C = Control, I = Information, P = Pricing

- IVHS = intelligent vehicle/highway system
- Varaiya, P. (1993). Smart cars on smart roads: problems of control. *IEEE Transactions on automatic control*, *38*(2), 195-207.

Pre-trip decisions

Trip planning: centralized vs. decentralized

Centralized

- A central command center sends instructions to each vehicle
- Optimize system-level performance e.g. average travel time & average fuel consumption
- Requires extensive information collection and communication capabilities
- Some win, some lose

Decentralized

- Drivers/vehicles plans trips independently, e.g. selfishly
- Optimize user-specific performance, e.g. travel time & fuel consumption
- Requires little communication and collaboration between vehicles
- Sometimes efficient, sometimes inefficient

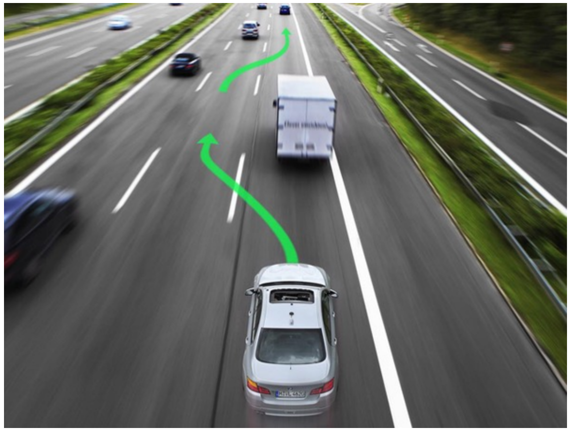
Trip planning: centralized vs. decentralized

- Reality is more likely to lie somewhere between fully centralized and fully decentralized.
- Information: central commander broadcast traffic information to vehicles, and vehicles may respond thereto
- Advice: central commander recommend decisions for vehicles, and vehicles may select whether to take
- Pricing: central commander charges those vehicles that take less favored actions
- Control: central commander forces vehicles to follow its commands via law enforcement capabilities

In-trip decisions

In-trip decisions

Route choice



Maneuver

Regulation

Route choice

- A better choice of route can reduce travel time.
- This choice can be improved by ‘offline’ information in the form of maps useful to drivers unfamiliar with the highway network, and by ‘online’ information about changes in traffic delays (caused by incidents or recurrent congestion), enabling the route to be adapted to changing traffic conditions.
- But simulation and analytical studies and data from demonstration experiments suggest little or no improvement from better choice of route under recurrent congestion and some improvement under non-recurrent (e.g., incident-induced) congestion.

Path planning

- Additional improvements are possible if instantaneous speeds are altered in ways that reduce or eliminate congestion.
- Evidence suggests that a small capacity increase, on the order of 15%, can be achieved by improving route choice and path planning decisions.
- For example, vehicles can intentionally decelerate to avoid congestion ahead.
- Also, vehicles typically have an optimal speed, and autonomous driving can track this speed.

Maneuvers

- Maneuvers refer primarily to the way drivers change lanes, including entry and exit from a highway.
- Maneuvers require the coordination of movement of neighboring vehicles.
- Improper coordination can result in congestion and accidents.
- In today’s unautomated traffic system, coordination is achieved by signaling (turn signals and brake lights) and social convention (e.g., providing room to a driver in the adjacent lane who indicates an intention to change lanes), buttressed by a legal code.

Maneuvers

- A partially automated system can improve maneuvers by providing a collision warning signal or a collision avoidance system which temporarily preempts driver action.

Regulation

- This primarily refers to the way a driver adjust the vehicle speed while keeping to one lane.
- A fundamental empirical fact about driver behavior is that, in steady state conditions, this speed is an increasing function of the distance from the vehicle in front.
- This also refers to how more complex maneuvers (e.g., merging & overtaking) can be implemented by computers.

Hardware requirements

Roadside Monitors

- They measure traffic conditions such as flow and speed.
- Based on these data the link layer calculates a path for each vehicle and the target platoon size and speed, and communicates them to the vehicle.
- Traffic measurements may be made by loop detectors, radar, or cameras.
- Information may be communicated to vehicles wireless communications (e.g., 5G).
- In case of centralized systems, high-speed computing is also required.

Hardware requirements

Onboard Devices:

- Vehicles must be equipped with a longitudinal sensor that measures the relative distance and speed between itself and the vehicle in front of it.
- Such sensors may be based on radar, lidar, or vision.
- In order to change lanes, the vehicle must be equipped with sensors that locate vehicles on the side within a range of about 30 m.
- The regulation layer also needs the vehicle's position and ground speed (which could be obtained in different ways) and several measurements of the state of the vehicle for which several sensors are available.
- High-speed onboard computer for perception & decision.

Hardware requirements

Inter-Vehicle Communication:

- The planning layer requires the ability to communicate with neighboring vehicles within a range of about 60 m.
- Such communication links need to be reliable, and incur very small delay (about 20 ms).
- Again, various solutions may be proposed including broadcast or cellular radio, infrared beacons, communication through a roadside station.
- There is significant progress in the development of systems for communicating routing and navigation data, but these systems do not meet the delay constraints needed for real time control.

Post-trip decisions

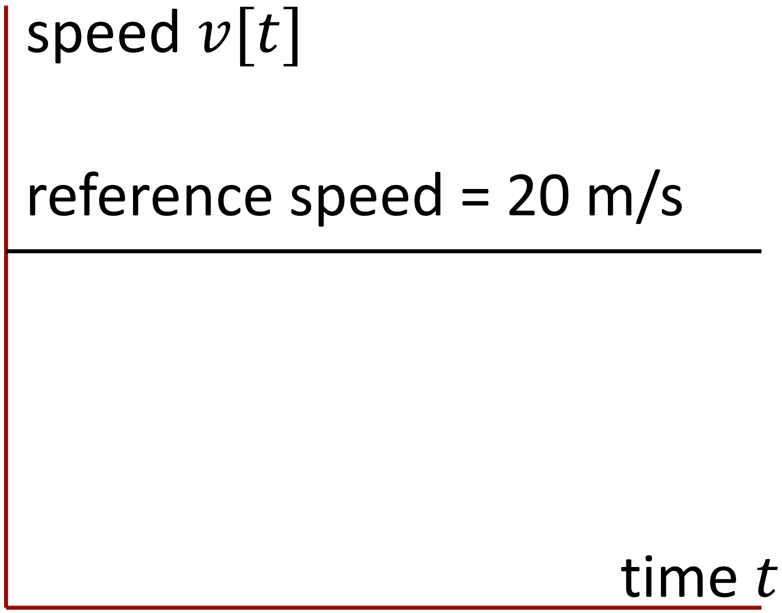
Outline

- Concept of autonomous driving
  - How human drive
  - How a computer drives
  - Decision-making architecture
- Speed tracking
  - State, dynamics, action, policy
  - Performance of policy
- 1D dynamical system & control
  - Modeling
  - Objective
  - Theory*

Control objective

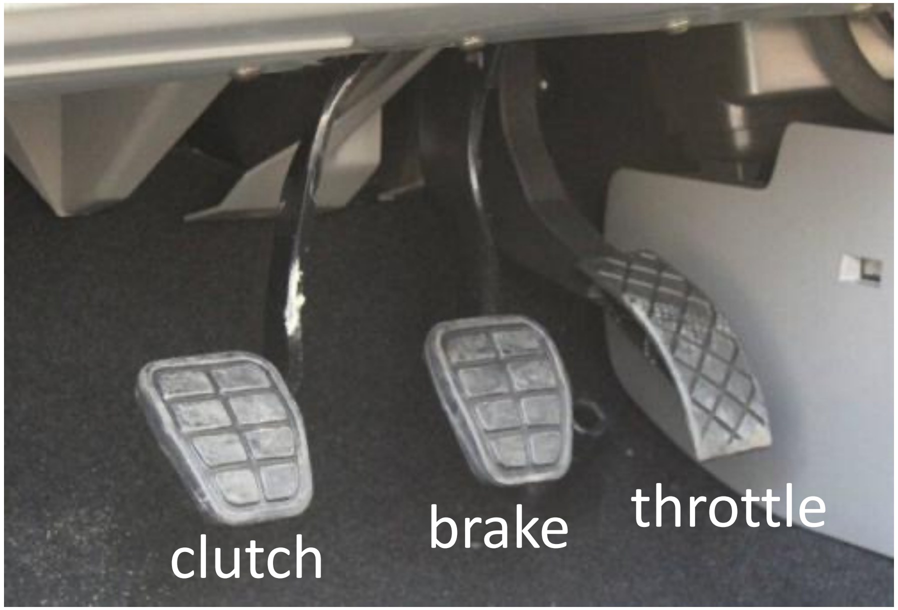
- We want a vehicle to travel at a particular speed
- Consider a single vehicle on a one-dimensional road.

- We want the vehicle to travel at a particular speed $\bar{v}$, say, 20 m/s. Called target/reference speed.
- We have a speedometer to observe the instantaneous speed.
- Assume no observation error.



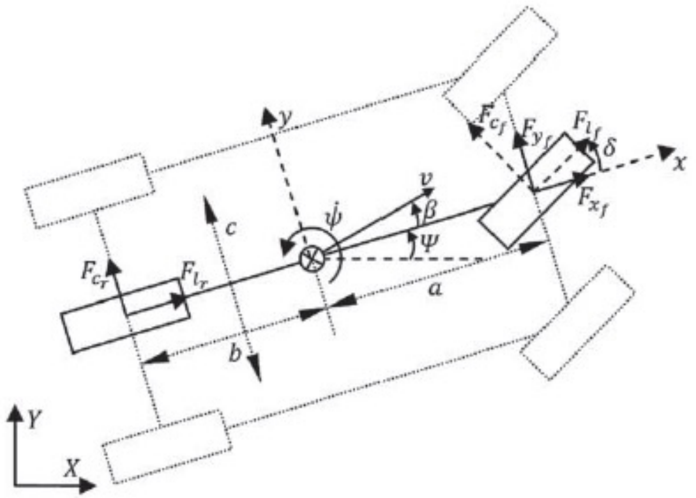
Action (control input)

- Note that the vehicle will not move unless a human driver or a computer tells it to do so.
- For linear motion, the action of a human/machine driver is to push/release the throttle/brake (if we assume automatic transmission without a clutch).



Action (control input)

- Throttle/brake influences the torque applied to the tires.
- The torque will determine the acceleration of the vehicle.



- For ease of presentation, we assume that we can directly determine the acceleration.
- Action (control input) $u[t]$ = acceleration at time t.

Dynamical system

- We can formulate the linear motion of a vehicle as a dynamical system as follows.
- State variable: $v[t]$ = speed at time t.
- State space: $\mathbb{R}_{\geq 0}$ = domain for state variable.
- Control input (action): $u[t]$ = acceleration at time t.
- System dynamics:

$$v[t+1] = v[t] + u[t]\delta.$$

- δ = discrete time step.
- We consider discrete times (DT) $0,1,2,\ldots$ for now.
- *To define a dynamical system, you need to specify (1) state, (2) control input, (3) system dynamics.*

Control objective

- Our primary objective is to select the sequence of control inputs $u[0], u[1], u[2], \ldots$ such that the vehicle travels at the desired speed.
- One way to formulate the above objective is: given an initial speed $v[0] = \nu \in \mathbb{R}_{\geq 0}$, there exists a time T such that $v[t] = \bar{v}$ for $t = T, T+1, T+2, \ldots$
- A weaker (yet more popular) formulation is
$$\lim_{t \to \infty} v[t] = \bar{v}.$$
- The latter is more popular, since actual engineering systems are always subject to errors and noises.
- You can never guarantee $v[t] = \bar{v}$ absolutely exactly.

Open-loop and closed-loop control

- Let's now consider two ways of selecting the control inputs.
- Assume zero initial speed, i.e., $v[0] = 0$.

1. Suppose that we want $v[t]$ to attain $\bar{v}$ after t_1 time steps. Thus, we can set
$$u[t] = \begin{cases} \dfrac{\bar{v}}{\delta t_1} & t = 0,1,\dots,t_1-1, \\ 0 & t = t_1, t_1+1, \dots \end{cases}$$
open loop
2. Suppose that we select $u[t]$ in response to the speed:
$$u[t] = \mu(v[t]), t = 0,1,2, \dots$$
closed loop

Open-loop control

- Idea of open-loop control:

  Specify $u[t]$ as a function of time t.
- In the context of vehicle control, this means you program a schedule of accelerations at every time step and send it to the vehicle.
- No more intervention after the schedule is sent.
- Send the instruction and let it go.

Open-loop control

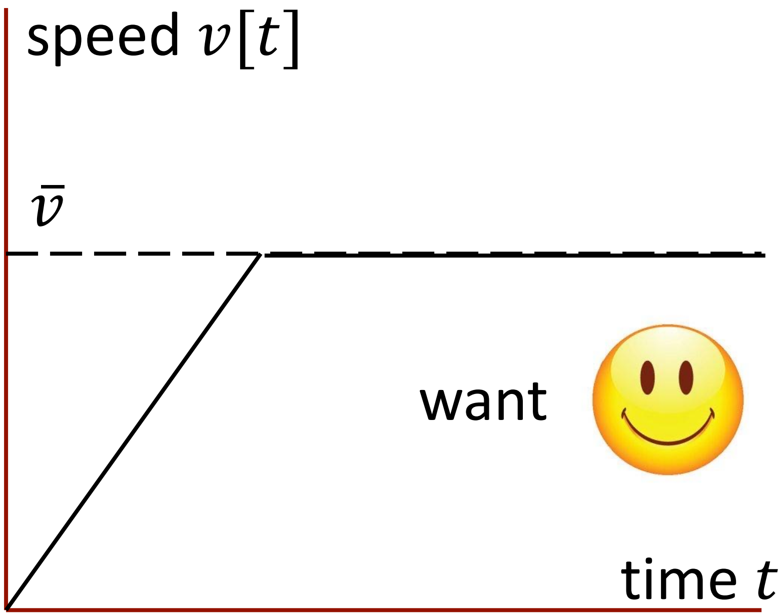
- This looks OK. What's the problem?
- Main reason: noise and error.
- Open-loop control is not valid unless <u>everything is perfect</u>.
- That is, our knowledge of the model is perfect, and the implementation of the control is perfect.
- Specifically, the time step size δ is absolutely perfect, and the acceleration $u[t]$ is exactly implemented.
- In practice, the above is never true!
- Time step size may fluctuate due to hardware/ communication error.

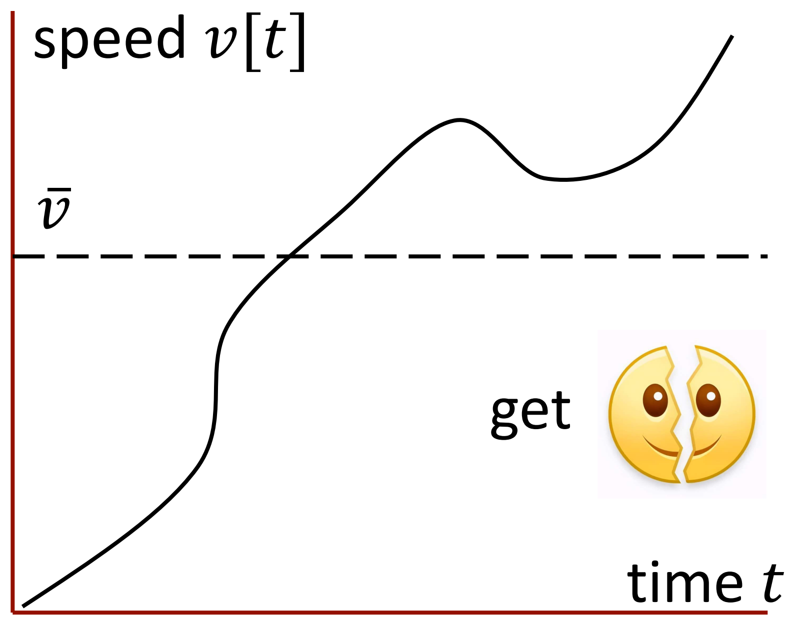
Open-loop control

- Implementation of $u[t]$ is never perfect.
- To attain the desired acceleration, we need to go (at least) through the following:
- *Push the pedal -> inject gas -> generate engine torque -> power transmitted to axis -> tire force translated to propelling force -> acceleration.*
- None of the above legs can be perfectly measured or implemented.
- In other words, in practice we usually have
$$v[t+1] = v[t] + u[t]\delta + w[t],$$
- $w[t]$ is a noise term capturing unmodeled factors.

Open-loop control

- What you want: $\lim_{t\to\infty} v[t] = \bar{v}$.
- What you get: $\lim_{t\to\infty} v[t] = \bar{v} + \sum_{t=0}^{\infty} w[t]$.
- Open-loop control does not provide any opportunity to prevent the noise from accumulating.



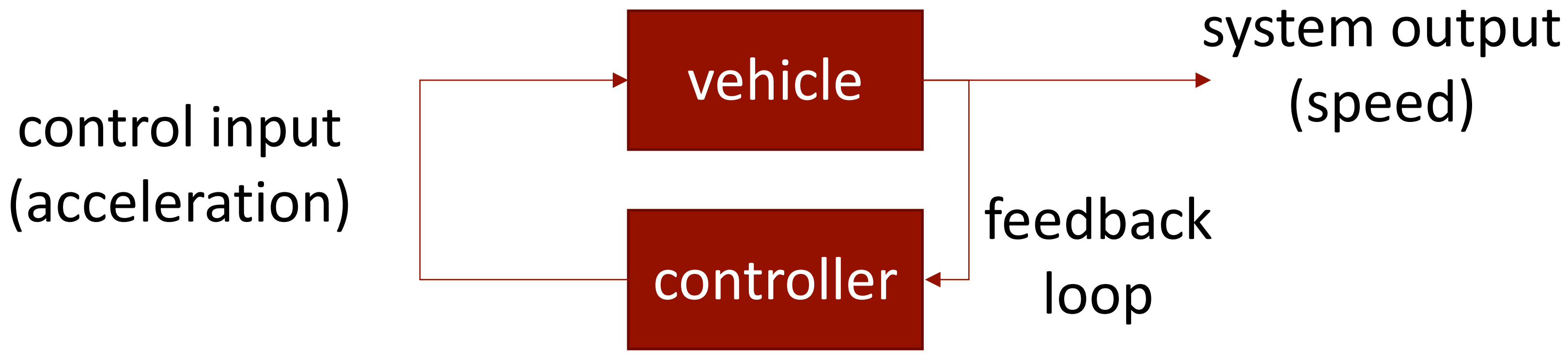


Closed-loop control

- Closed-loop control

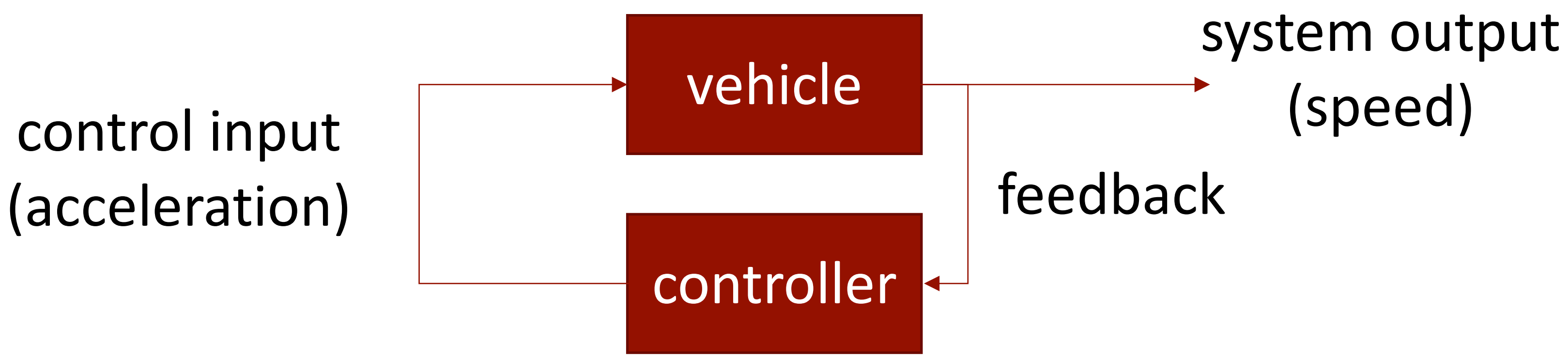
Specify $u[t]$ as a function of state $v[t]$.

- In the context of vehicle control, this means that at every time step you select the acceleration according to the current speed.
- Persistent intervention as time goes.
- This is called closed-loop control.



Closed-loop control

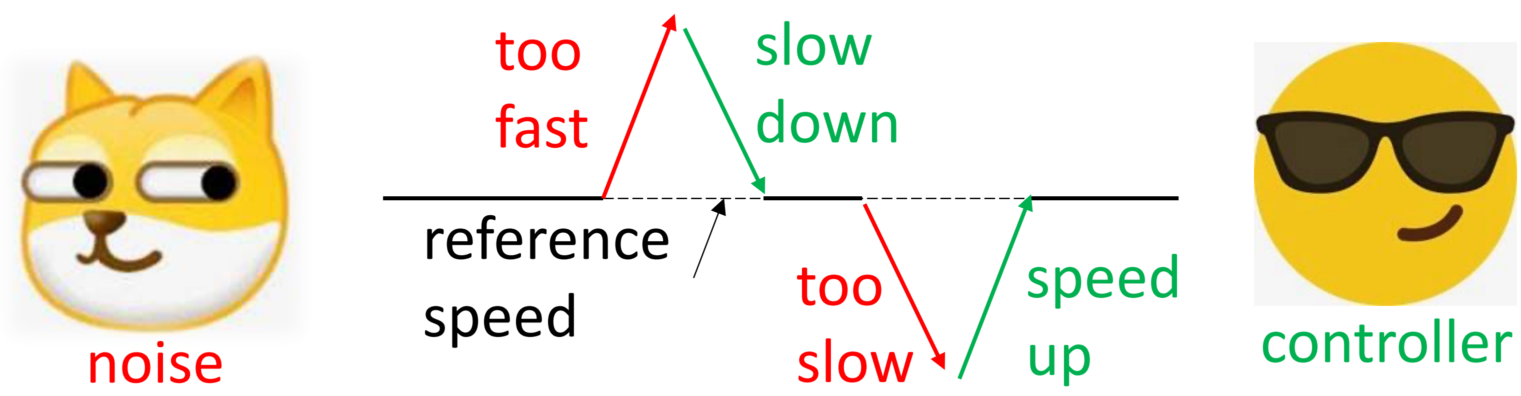
- Closed-loop control is able to compensate for error and noise via feedback.



- Key idea: controller compares actual speed (output) with the desired speed (reference).
- If vehicle is faster than specified, then slow it down.
- If vehicle is slower than specified, then speed it up.
- How to quantify such intuition!

Closed-loop control

- With feedback, we are no longer afraid of noises.
- Whenever noise deviates the speed, feedback controller will steer the speed back to the reference.
- No error accumulation.



Closed-loop control

- Essentially, we need a mapping from state (speed) to control input (acceleration).
- That is, I tell you how fast the vehicle is traveling, you must tell me how much acceleration should be applied.
- It’s almost like looking up a table:

Speed [m/s]	Acceleration [m/s^2]
0	?
1	?
2	?
3	?
⋮	⋮

Control policy

- This mapping is called the control policy.
- Also called control law or controller or simply policy.
- Typically, the mapping is a function.
- (You may want to recall the difference between a mapping and a function).
- Every one is supposed to know the following:

1. **A control policy is a function;**
2. **This function maps a state to a control input.**

- Learn the above by heart!
- Even some automation-majored students are confused.

Controlled dynamics

- Suppose that we apply a controller
$$\mu: \mathbb{R} \to \mathbb{R}.$$
$$\mu: v \mapsto u.$$
- Maps speed to acceleration.
- Dynamics of the feedback-controlled system:
$$v[t+1] = v[t] + \mu(v[t])\delta + w[t].$$
- Thus, the right-hand side (RHS) is essentially a function of $v[t]$.
- When the noise $w[t]$ deviates the speed, the controller μ compensates such deviation.

Linear controller

- Suppose that $\mu(v) = k(v - \bar{v})$.
- If you observe the dynamical equation
$$v[t+1] = v[t] + u[t]\delta,$$
you will see that k has to be negative.
- That is, if $v[t] > \bar{v}$, then $u[t] < 0$: vehicle decelerates if traveling too fast.
- If $v[t] < \bar{v}$, then $u[t] > 0$: vehicle accelerates if traveling too slow.
- One can further see that
$$v[t+1] = v[t] + k(v[t] - \bar{v})\delta = (1 + k\delta)v[t] - k\delta\bar{v}.$$
- Fixed point $v = \bar{v}$: $\bar{v} = (1 + k\delta)\bar{v} - k\delta\bar{v}$.

Outline

- Concept of autonomous driving
  - How human drive
  - How a computer drives
  - Decision-making architecture
- Speed tracking
  - State, dynamics, action, policy
  - Performance of policy
- 1D dynamical system & control
  - Modeling
  - Objective
  - Theory*

State & control

- Consider a DT one-dimensional dynamical system.
- State variable: x. (Notational convention)
- State space: $\mathbb{R}$.
- $x[t] \in \mathbb{R}$ for $t = 0,1,2, \ldots$
- Control input: u. (Again, notational convention)
- Set of inputs: $\mathbb{R}$.
- $u[t] \in \mathbb{R}$ for $t = 0,1,2, \ldots$
- Note that $x[t]$ and $u[t]$ are sufficient for predicting future evolution; $x[t-1], x[t-2]$ no longer matters.

Dynamics

- Initial condition: $x[0] \in \mathbb{R}$.
- Dynamical equation:
$$x[t+1] = f(x[t], u[t]).$$
- If the function $f: \mathbb{R}^2 \to \mathbb{R}$ is linear in its arguments, then the system is linear.
- That is, the system is linear if f takes the form
$$f(x, u) = ax + bu + c.$$
- Typically we assume $c = 0$.
- Then, we have
$$x[t+1] = ax[t] + bu[t].$$
- Such a system is called linear time-invariant (LTI).

Non-LTI systems

- A system is not LTI if it is either nonlinear or time-varying.
- If the function $f(x, u)$ is nonlinear in x and u, the system is nonlinear.
- For example, the following system is nonlinear:
$$x[t+1] = ax[t]u[t].$$
- If the function $f(x, u; t)$ depends on time t, the system is time-varying.
- For example, the following system is time-varying:
$$x[t+1] = (a_0 + a_1 t)x[t] + bu[t].$$
- Non-LTI systems are significantly more complex than LTI ones.

Control of LTI systems

- Consider a 1D LTI system
$$x[t+1] = ax[t] + bu[t].$$
- A typical control objective is to steer the system to a desired state.
- To make math simpler, we usually set the desired state to be 0.
- This can be done by shifting the origin of the state space $\mathbb{R}$.
- Hence, the control problem is: given initial condition $x[0]$, select $u[t]$ for $t = 0,1,2, \ldots$ such that
$$\lim_{t \to \infty} x[t] = 0. \text{ (Asymptotic convergence)}$$

Feedback control

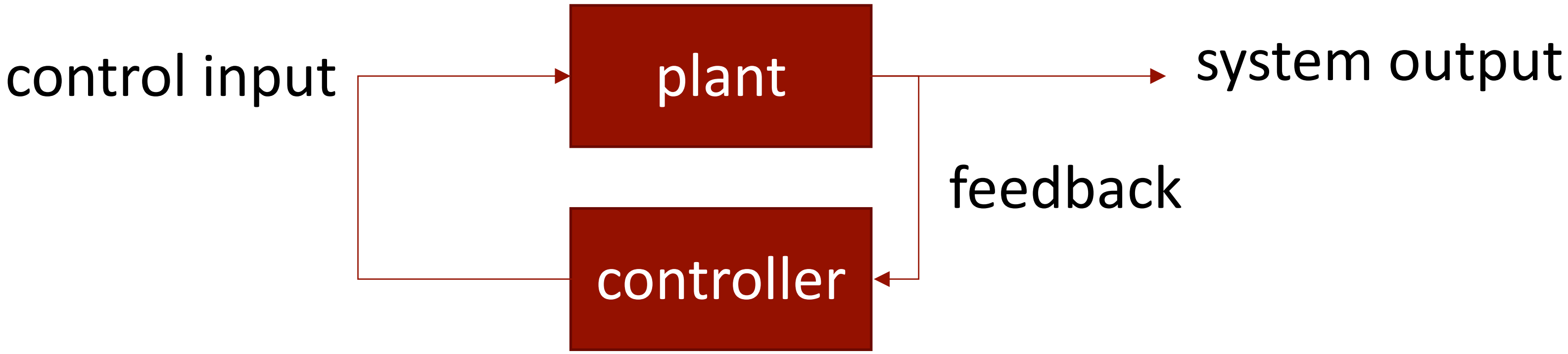
- Feedback control means to select $u[t]$ according to $x[t]$.
- Mathematically, we look for a function $\mu: \mathbb{R} \to \mathbb{R}$ such that
$$u[t] := \mu(x[t]), \qquad t = 0,1,2, \ldots$$
- The function μ is called

control policy/control law/controller.

- If the control policy is linear, i.e., if
$$\mu(x) = kx,$$
the feedback-controlled system is also linear:
$$x[t+1] = ax[t] + bkx[t] = (a+bk)x[t].$$

Use of feedback control

- Feedback (i.e., closed-loop) control is able to compensate for disturbances/noises that deviate the system from its predicted evolution.
- Open-loop control is unable to do the above.
- However, feedback control is more costly than open-loop control in that you need to observe the state.
- This involves investment in sensing capabilities.



CT LTI system

- Sometimes we also consider CT LTI systems:

$$\frac{d}{dt}x(t) = ax(t) + bu(t) \text{ or simply } \dot{x} = ax + bu.$$

- To obtain the DT counterpart, consider

$$\begin{aligned} x(t+\delta) &= x(t) + \big(ax(t) + bu(t)\big)\delta \\ &= (1 + a\delta)x(t) + b\delta u(t). \end{aligned}$$

- This is called discretization.
- Key: only $t + \delta$ on the left hand side, only t on the right hand side.
- If a linear controller is applied,

$$\frac{d}{dt}x(t) = ax(t) + bkx(t).$$

Criterion for convergence

Assuming real-valued parameters.

DT: $x[t+1] = ax[t] + bkx[t] = (a+bk)x[t]$

$$x[t] = (a+bk)^t x[0], \qquad t = 1,2,\ldots$$

- **Criterion for convergence**: $|a+bk| < 1.$

CT: $\dot{x}(t) = ax(t) + bkx(t) = (a+bk)x(t)$

$$x(t) = x(0)e^{(a+bk)t}, \qquad t > 0.$$

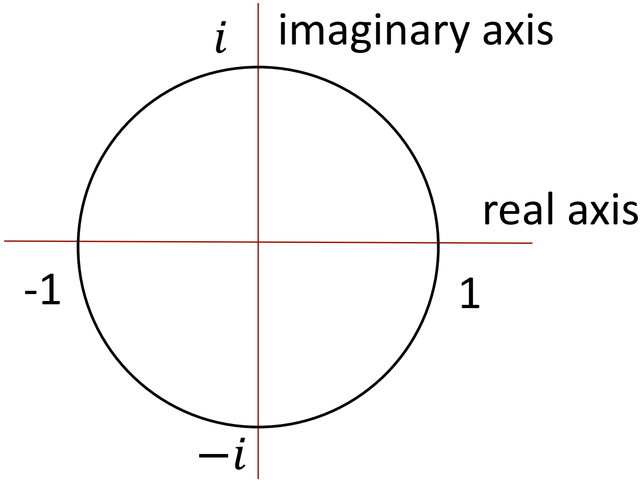
- **Criterion for convergence**: $a+bk < 0.$

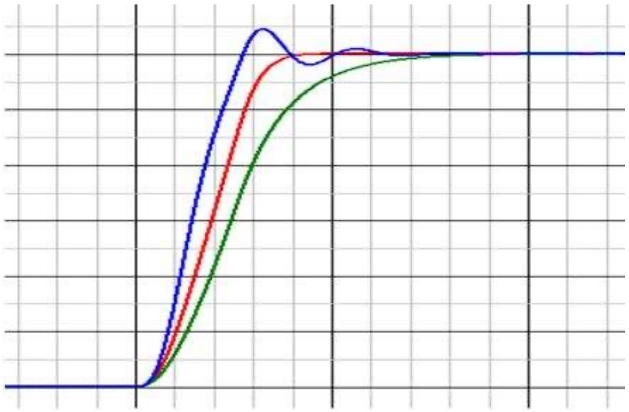
[Not required] How to select $\boldsymbol{k}$ (DT)

- Feedback controller:
$$\mu(x) = kx.$$
- Thus, we have
$$x[t] = (a + bk)^t x[0], \qquad t = 0,1,2, \dots$$
- Therefore, the system is convergent in the sense that $\lim_{t \to \infty} x[t] = 0$ if and only if
$$|a + bk| < 1.$$
- That is, the magnitude of $a + bk$ is strictly less than 1.
- What if $a + bk$ is complex-valued?
- Recall: complex numbers = real part + imaginary part
- Imaginary unit: $i = \sqrt{-1}$.

[Not required] How to select $\boldsymbol{k}$ (DT)

- Note that $a + bk$ does not have to be real.
- $|a + bk| < 1$ means that $a + bk$ must fall in the interior of the unit circle on the complex plane.





- If $|a + bk| < 1$ and if $a + bk$ is real, then $x[t]$ ”smoothly” decreases to 0.
- If $|a + bk| < 1$ and if $a + bk$ is complex, then $x[t]$ converges to 0 with oscillations.

[Not required] How to select $\boldsymbol{k}$ (CT)

- Consider the CT closed-loop system
$$\dot{x} = ax + bkx = (a + bk)x.$$
- The solution is given by
$$x(t) = x(0)e^{(a+bk)t}, \qquad t > 0.$$
- Hence, the system is convergent if and only if
$$\mathrm{Re}(a + bk) < 0.$$
- That is, the real part of $a + bk$ is strictly negative.
- In other words, $a + bk$ has to be on the left half plane on the complex plane.
- Again, non-zero imaginary part of $a + bk$ leads to oscillation. (Recall $e^{i\theta} = \cos\theta + i\sin\theta$.)

Summary

- Concept of autonomous driving
  - How human drive
  - How a computer drives
  - Decision-making architecture
- Speed tracking
  - State, dynamics, action, policy
  - Performance of policy
- 1D dynamical system & control
  - Modeling
  - Objective
  - Theory*