5. Autonomous Driving: Vehicle platooning

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

- Longitudinal control
  - Dynamical equation
  - State-space model
- Synthesis with tracking & following algorithms
  - Trajectory tracking
  - Vehicle following
- Additional issues
  - Linearization
  - Saturation
  - Noise & perturbation
  - Model identification
  - Human driver behavior

Outline

- Technological basis
  - Connected and autonomous vehicles
  - Vehicle platooning
- Simplified formulation
  - Modeling
  - Decision making
  - Final project option 1
- State-of-the-art formulation
  - Modeling
  - Decision making
  - String stability

- Ref: Alam et al. Heavy-Duty Vehicle Platooning for Sustainable Freight Transportation.

Adaptive cruise control (ACC)

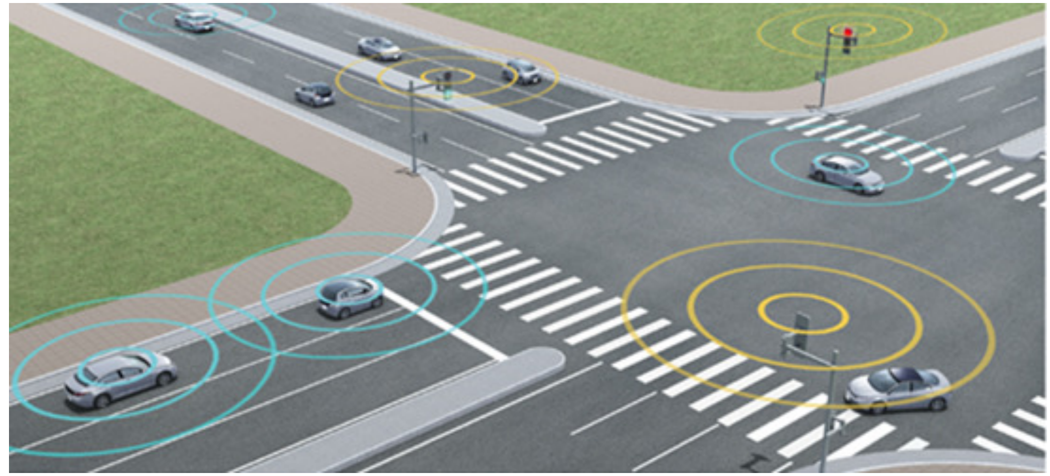
- Longitudinal: vehicle following
- Recall lecture 4
- Lateral: mostly lane keeping; sometimes lane changing

Cooperative adaptive cruise control (CACC)

- CACC: 协同自适应巡航控制
- Two key words:
  - Cooperative（协同的）: multiple vehicles share information and jointly make decisions
  - Adaptive（自适应的）: control inputs are generated in response to real-time condition
- CACC drives better than human, since
  - vehicles talk to each other and can proactively and anticipatorily account for the behavior of other vehicles;
  - computers can respond faster than human
- Note: CACC still needs a human driving sitting in the vehicle for <u>monitoring</u> & <u>higher-level decisions</u>.

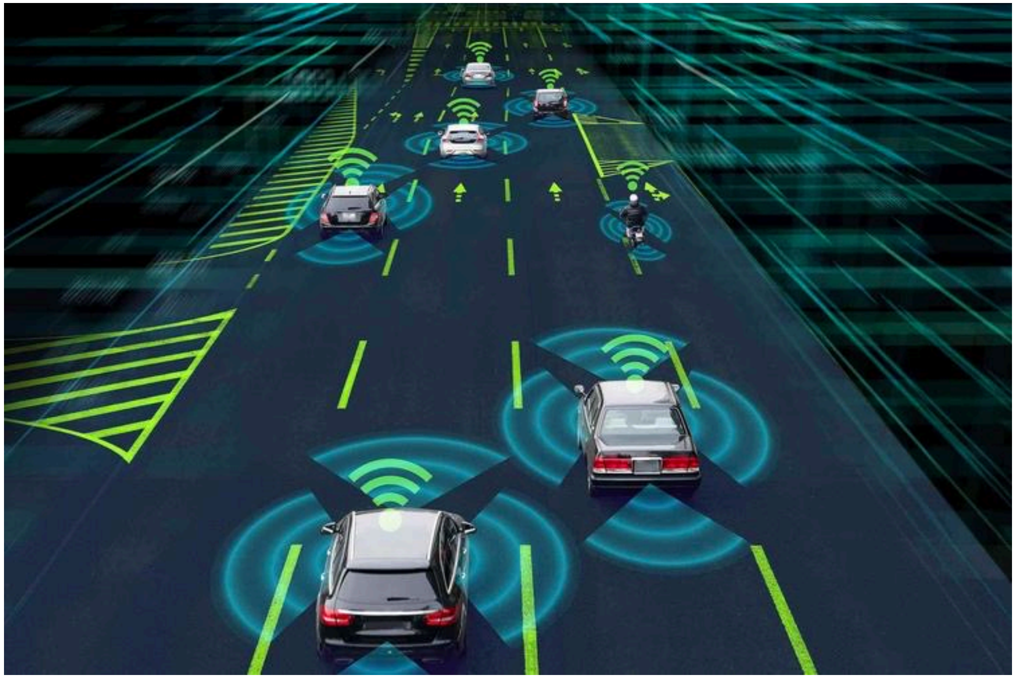
Vehicle-to-vehicle coordination

- Onboard unit (OBU)
- Broadcast information to neighboring vehicles:
  - Vehicle type
  - **Latest** position, speed, acceleration, orientation…
  - **Intended** position, speed, acceleration, orientation…



Connected and autonomous vehicle (CAV)

- Connected: real-time information exchange between vehicles.
- Autonomous: autonomous driving or adaptive cruise control.



Platooning

https://www.bilibili.com/video/BV1nJ411h7Tb?spm_id_from=333.337.search-card.all.click

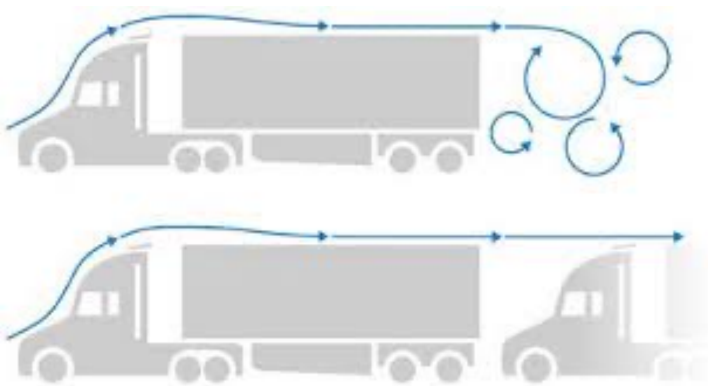
Platooning: Motivation

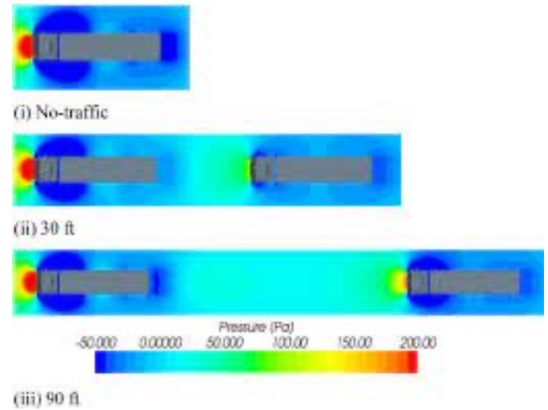
- Limited highway capacity -> more traffic jams
- A naïve solution: build wider roads.
- A smarter solution: reduce the intervehicle distance.

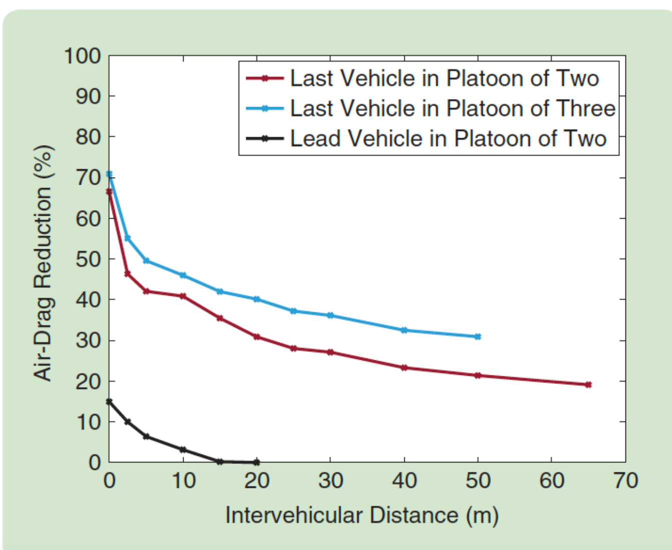
- Before the age of AI, human could not do that.
- But now, computers can do that! (time gap: 2s ->0.5s)

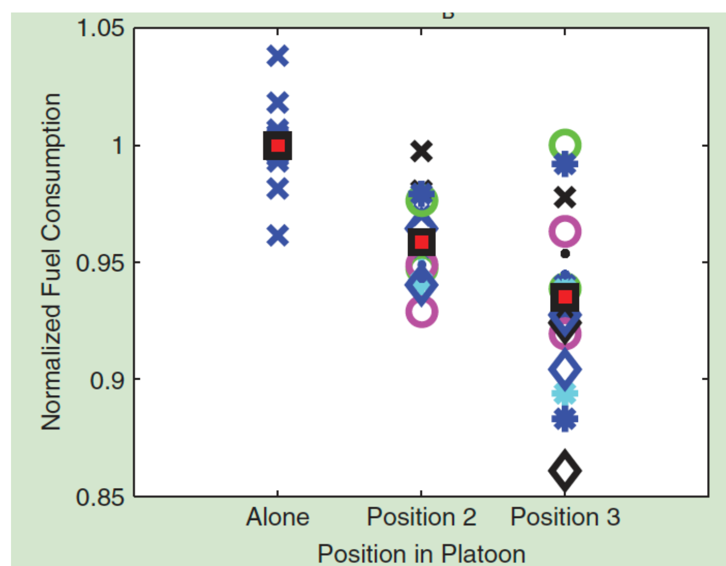
Platooning: Motivation

- Reduces air drag; saves fuel (up to 15%)









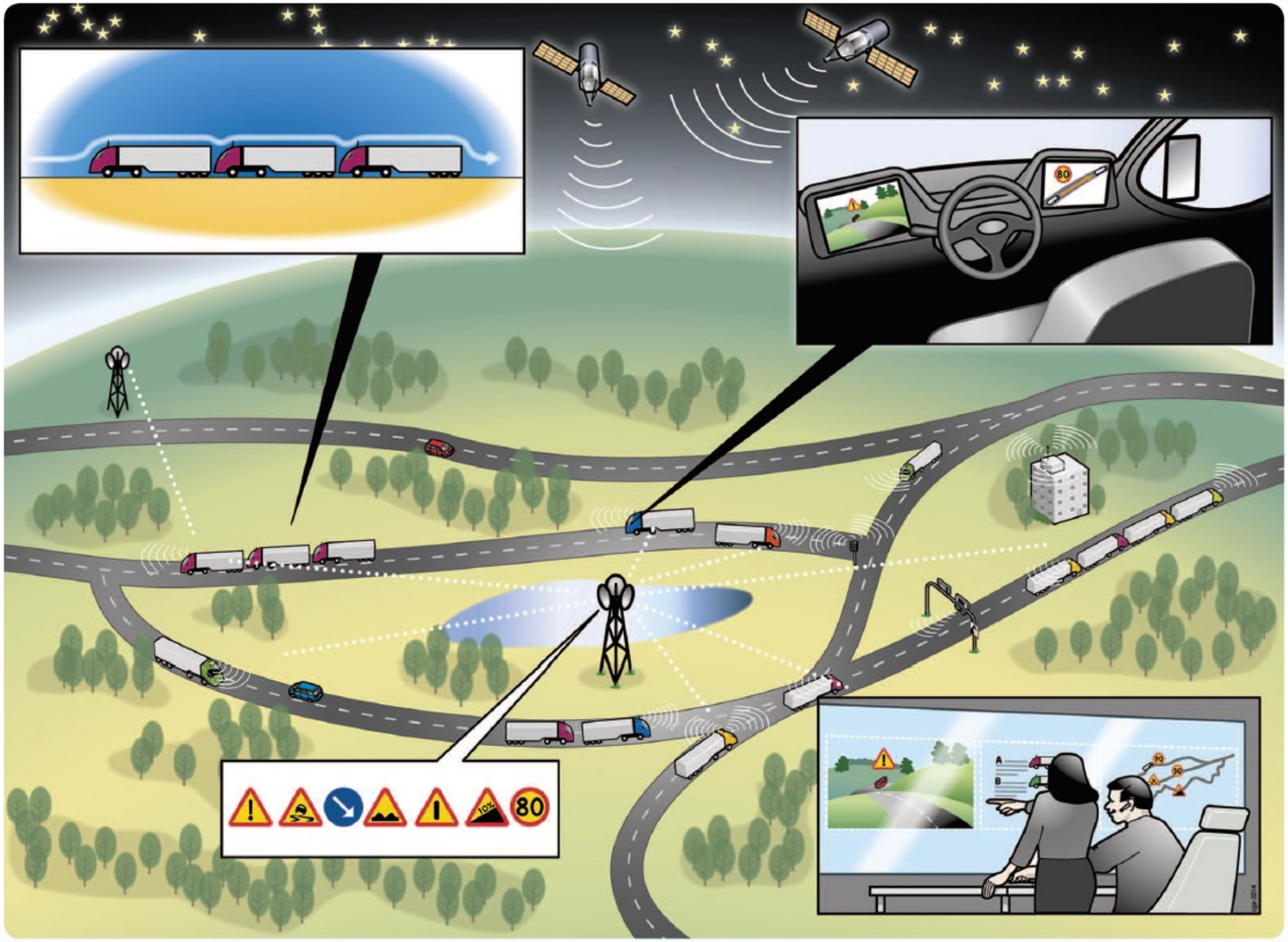
Platooning: Motivation

- Improved safety & working condition for drivers.
- Automatic systems can usually react more quickly to dangerous situations and can exploit additional information resulting from the communication and cooperation between vehicles.

Platooning: three layers

- A hierarchical decomposition of the overall transportation problem into distinct layers.
- Three-layer architecture: transportation, platoon, and vehicle layers.
- Transport layer is responsible for transport planning (that is, assigning goods to vehicles) and vehicle routing.
- Platoon layer translates the desired route into a specific trajectory for each vehicle, including platooning maneuvers such as the merging or splitting of platoons.
- Vehicle layer is aimed at tracking the desired trajectories from the platoon by real-time vehicle control.

Platooning: three layers



Platooning: three layers

Transportation layer:

- The transportation layer handles the transport planning problem by distributing the required flow of goods/passengers over the available vehicles and subsequently assigning their routes.
- Thus, the transport layer comprises two closely related tasks: transport planning and vehicle routing.
- The objective of the transport planning task is to maximize the capacity utilization of vehicles by grouping similar transport assignments into a single load.
- More of an optimization rather than a control problem.

Platooning: three layers

Platoon Layer

- The platoon layer takes the planned routes and platooning schedules, as computed by the transport layer, and assigns a reference velocity profile for each vehicle.
- This results in two distinct tasks: look-ahead trajectory planning and execution of platooning maneuvers.
- Typical objective: using minimal fuel to cover a required trip.
- Recall trajectory tracking problem.

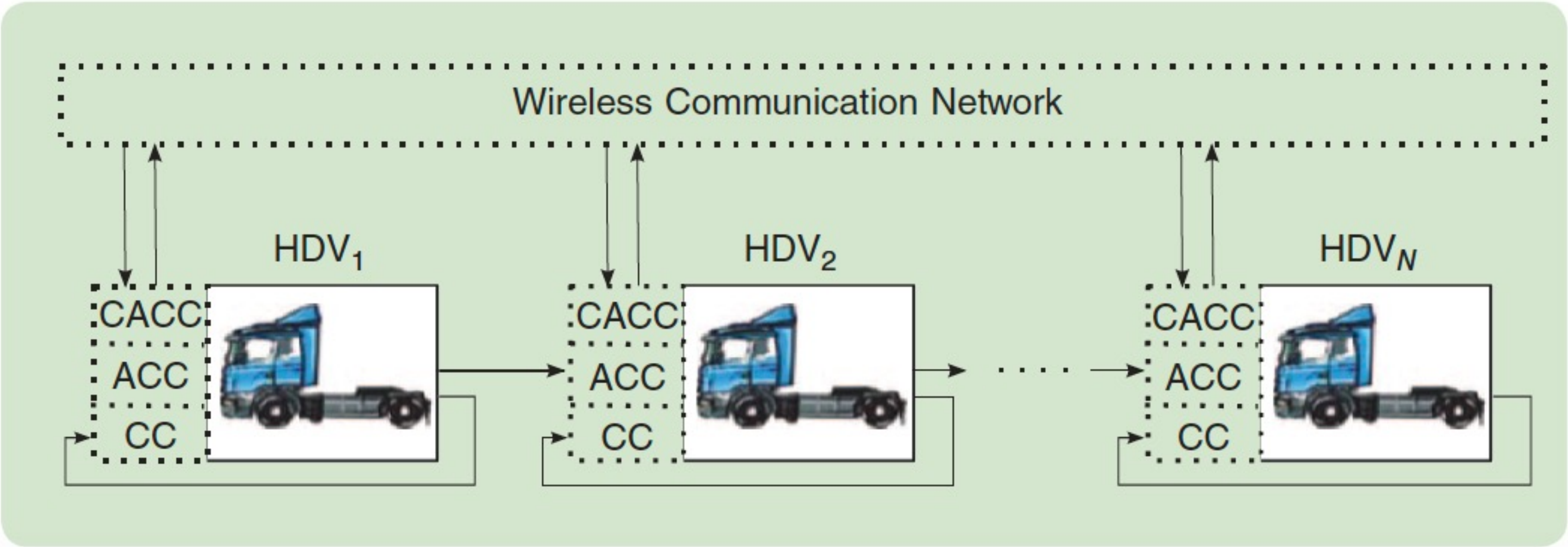
Platooning: three layers

Vehicle Layer (focus of this lecture)

- The vehicle layer deals with the real-time control of individual vehicles, using the output from the platoon layer as a reference trajectory.
- The onboard vehicle controller ensures tracking of the desired velocities and inter-vehicular distances, exploiting V2V communication and (radar) measurements of the inter-vehicular distance.
- This controller should ensure a proper rejection of local disturbances, in which the concept of string stability is important.

Platooning: Architecture

- A platoon system architecture for an N-vehicle platoon.
- The information flow over communication and sensor channels is illustrated by the arrows.
- The control options for vehicle speed control are shown in front of each vehicle.



Platooning: Architecture

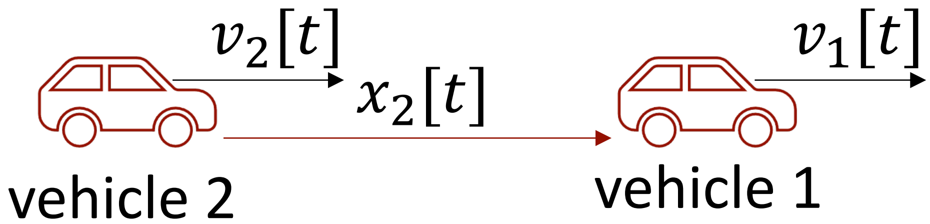
- For platoon driving, the first vehicle will use either a cruise controller (CC) or adaptive cruise controller (ACC).
- The follower vehicles will employ the cooperative adaptive cruise controller (CACC) when a wireless connection is established.
- The first vehicle might switch to the CACC controller when approaching another platoon.
- The cruise control (CC) and ACC are commercially available systems in most modern vehicles, especially heavy-duty vehicles (HDVs).

Outline

- Technological basis
  - Autonomous driving
  - Vehicle-to-vehicle coordination
- Simplified formulation
  - Modeling
  - Decision making
  - Final project option 1
- State-of-the-art formulation
  - Modeling
  - Decision making
  - String stability

Architecture of platooning

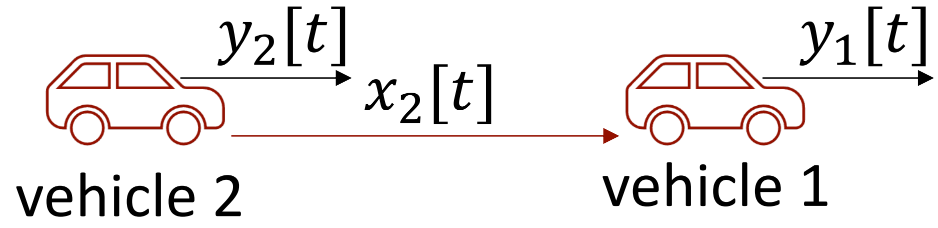
- Consider two CAVs.



- Vehicle 1 does a trajectory tracking problem.
- Vehicle 2 does a vehicle following problem.
- Two methodologies for vehicle 2:

1. ACC: vehicle 2 only knows vehicle 1’s state but does not know vehicle 1’s action.
2. CACC: vehicle 2 knows vehicle 1’s both state and action.

State & control



State variables:

- Vehicle 1:
  - $x_1 \in \mathbb{R}$: deviation from reference position
  - $v_1 \in \mathbb{R}$: speed; or $y_1 \in \mathbb{R}$: relative speed w.r.t. reference speed.
- Vehicle 2:
  - $x_2 \in \mathbb{R}$: deviation from reference position (see below)
  - $v_2 \in \mathbb{R}$: speed; or $y_2 \in \mathbb{R}$: relative speed w.r.t. reference

Control objective:

- Speed: we want both vehicles to travel at speed $\bar{v}$.
- Spacing: we want a spacing of d between two vehicles.

Control of vehicle 1

- State variable: $x_1[t], y_1[t]$
- State space: $\mathbb{R}^2$
- Control input: $u_1[t]$ = acceleration; input space = $\mathbb{R}$
- Dynamical equation

$$\begin{bmatrix} x_1[t+1] \\ y_1[t+1] \end{bmatrix} = \begin{bmatrix} 1 & \delta \\ 0 & 1 \end{bmatrix} \begin{bmatrix} x_1[t] \\ y_1[t] \end{bmatrix} + \begin{bmatrix} 0 \\ \delta \end{bmatrix} u_1[t].$$

- Control objective: asymptotic convergence
- Control policy:

$$\mu_1(x_1, y_1) = -k_{11}x_1 - k_{12}y_1.$$

- Control design task: select k_{11}, k_{12}.
- Equilibrium state: $x_1 = 0, y_1 = 0$.

Control of vehicle 1

- Recall from longitudinal dynamics:
$$\dot{v}_1 = -\frac{\rho C_d}{2M_t} v^2 + \frac{T_{1e} - R_g T_{1b}}{M_t} - \frac{C_r m g}{M_t}.$$
- Hence, we can determine the torques $T1_e$ and T_{1b} according to the acceleration $u_1 = \mu_1(x_1, y_1)$.
- Indeed, in practice we also need to deal with lateral control.
- In any case, vehicle 2 will not influence the motion of vehicle 1.
- Vehicle 1 moves as if vehicle 2 does not exist.

Control of vehicle 2

- State variable: $x_2[t], y_2[t]$
- $x_2[t] = 0$ if exactly d away from vehicle 1.
- Control input: $u_2[t]$ = acceleration; input space = $\mathbb{R}$
- Note that $x_2[t+1] = x_2[t] + y_2[t]\delta$.
- Dynamic equation

$$\begin{bmatrix} x_2[t+1] \\ y_2[t+1] \end{bmatrix} = \begin{bmatrix} 1 & \delta \\ 0 & 1 \end{bmatrix} \begin{bmatrix} x_2[t] \\ y_2[t] \end{bmatrix} + \begin{bmatrix} 0 \\ \delta \end{bmatrix} (u_2[t] - u_1[t]).$$

- Equilibrium state: $x_2 = 0, y_2 = 0$.
- That is, spacing = d and speed = $\bar{v}$.
- We proceed as if this is trajectory tracking.

Comparison between platooning & following

Vehicle following:

- Based on ACC; no collaboration.

$$\begin{bmatrix} x_2[t+1] \\ y_2[t+1] \end{bmatrix} = \begin{bmatrix} 1 & \delta \\ 0 & 1 \end{bmatrix} \begin{bmatrix} x_2[t] \\ y_2[t] \end{bmatrix} + \begin{bmatrix} 0 \\ \delta \end{bmatrix} (u_2[t] - u_1[t])$$

- Need to predict/estimate $u_1[t]$.
- Typically assume a worst-case value to ensure safety.

Vehicle platooning

- Based on CACC; with collaboration.

$$\begin{bmatrix} x_2[t+1] \\ y_2[t+1] \end{bmatrix} = \begin{bmatrix} 1 & \delta \\ 0 & 1 \end{bmatrix} \begin{bmatrix} x_2[t] \\ y_2[t] \end{bmatrix} + \begin{bmatrix} 0 \\ \delta \end{bmatrix} (u_2[t] - u_1[t])$$

- Vehicle 2 knows $u_1[t]$; avoids over-conservatism.

Control of vehicle 2: centralized

- By lumping the models for both vehicles, we can obtain a linear system with state $\left[x_1[t], y_1[t], x_2[t], y_2[t]\right]^T$ and control $\left[u_1[t], u_2[t]\right]^T$:

$$\begin{bmatrix} x_1[t+1] \\ y_1[t+1] \\ x_2[t+1] \\ y_2[t+1] \end{bmatrix} = \begin{bmatrix} 1 & -\delta & & \\ & 1 & & \\ & & 1 & \delta \\ & & & 1 \end{bmatrix} \begin{bmatrix} x_1[t] \\ y_1[t] \\ x_2[t] \\ y_2[t] \end{bmatrix} + \begin{bmatrix} & \\ \delta & \\ & \\ -\delta & \delta \end{bmatrix} \begin{bmatrix} u_1[t] \\ u_2[t] \end{bmatrix}$$

Control of vehicle 2: centralized

- Then, design a linear controller such that
$$\begin{bmatrix} u_1[t] \\ u_2[t] \end{bmatrix} = K \begin{bmatrix} x_1[t] \\ y_1[t] \\ x_2[t] \\ y_2[t] \end{bmatrix}.$$
- You can incorporate the controller designed for vehicle 1 into the matrix K.
- You can also assume no influence of vehicle 2 on the control of vehicle 1.
- Then, you can select the coefficients associated with u_2 to obtain a stabilizing controller.
- This scheme may not work for multi-vehicle platoons...

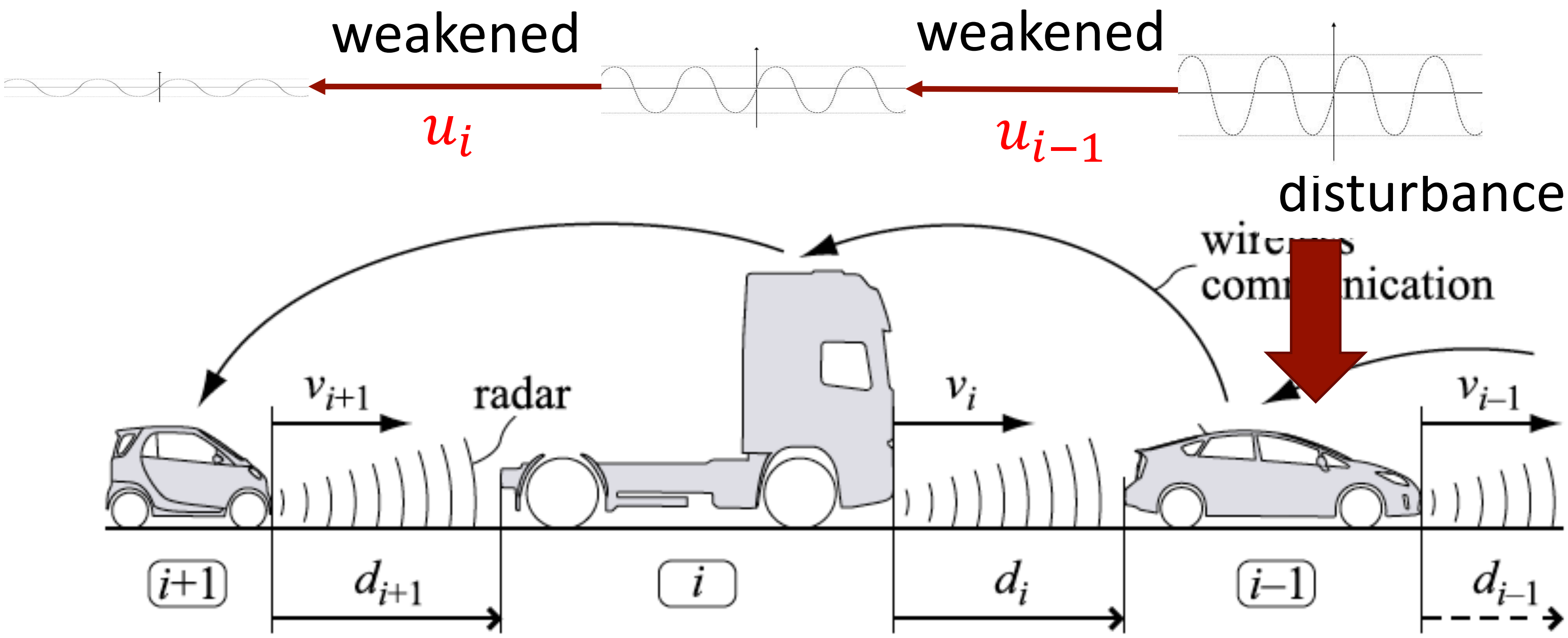
Control of vehicle 2: decentralized

- For a two-vehicle platoon, a decentralized control scheme means that

$u_2[t]$ does not immediately depend on $x_1[t]$ and $y_1[t]$.

- That is, u_2 should be given by
$$u_2 = \mu_2(x_2, y_2) = -k_{21}x_2 - k_{22}y_2.$$
- Then, the dynamical equation for vehicle 2 will depend on vehicle 1.
- How can you ensure that vehicle 2 is converging?
- Use the notion of string stability.

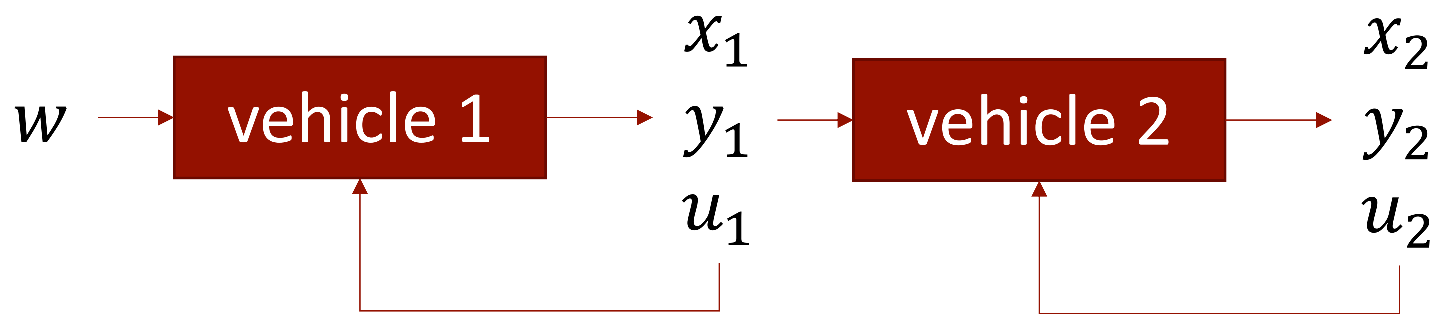
String stability



- String stability: any disturbance you impose on a vehicle will be weakened as its impact propagates to downstream vehicles.
- Specifically, u_i should weaken disturbances in u_{i-1}.

String stability

- Suppose that $u_1[t]$ is perturbed by w.



- First, the impact of w will keep circling in vehicle 1's control loop.
- Second, the impact of w will be transmitted to vehicle 2 and keep circling in vehicle 2's control loop.
- Is this OK?

Final project option 1

Compare ACC with CACC in the face of at least three out of the following complications:

Saturation, Noise, Modeling error, Human behavior, Time-varying environment...

- Required: formulation & simulation.
  - Presentation of the state-space model & the control policy.
  - Explanation of incorporation of the above complications.
  - Simulation-based comparison, validation, and optimization.
- Not required: theoretical analysis.
  - Proof of convergence of proposed control policy.
  - Formulation of optimal control problem.

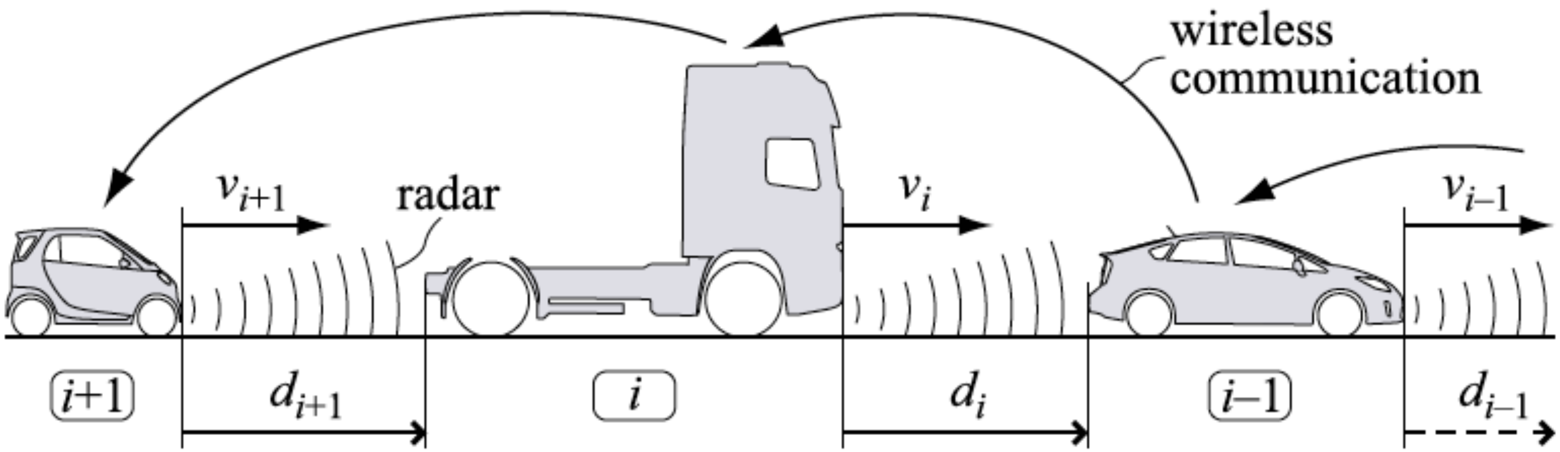
Outline

- Technological basis
  - Autonomous driving
  - Vehicle-to-vehicle coordination
- Simplified formulation
  - Modeling
  - Decision making
  - Final project option 1
- State-of-the-art approach
  - Modeling
  - Decision making
  - String stability

Ref: Ploeg, J., Van De Wouw, N., & Nijmeijer, H. (2013). L_p string stability of cascaded systems: Application to vehicle platooning. *IEEE Transactions on Control Systems Technology*, *22*(2), 786-793.

Platoon dynamics

- Consider a platoon of m vehicles
- d_i = the distance between vehicle i and its preceding vehicle $i - 1$
- v_i its velocity.
- The objective of each vehicle is to follow the preceding vehicle at a desired distance $d_{r,i}$



Reference spacing

- The objective of each vehicle is to follow the preceding vehicle at a desired reference spacing $d_{r,i}$

$$d_{r,i}(t) = r_i + hv_i(t), \qquad i \in S_m$$

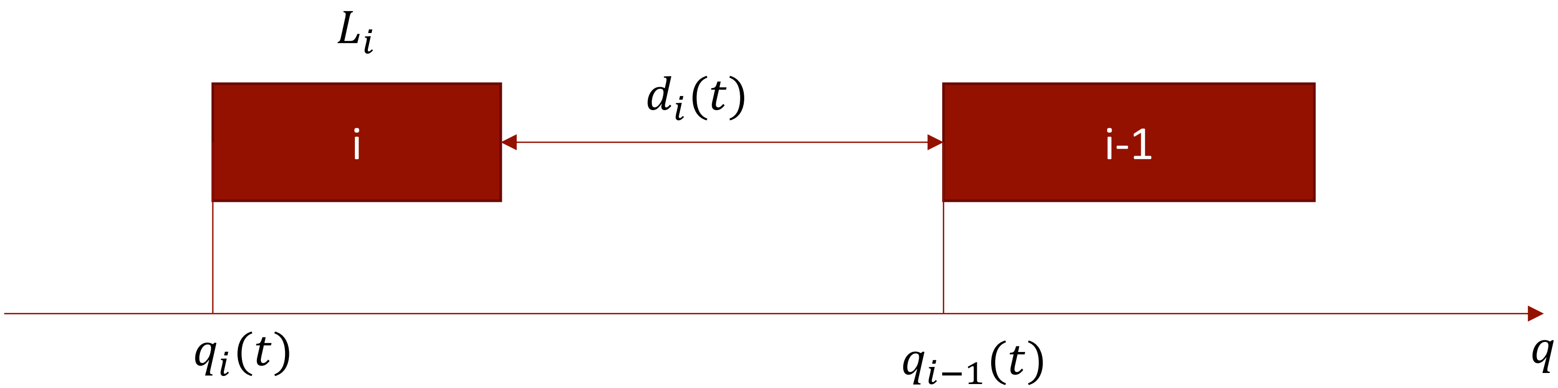
- h = the time headway (assuming homogeneous platoon)
- r_i = the standstill distance.
- $S_m = \{i \in N \mid 1 \leq i \leq m\}$ is the set of all vehicles in a platoon of length $m \in \mathbb{N}$.
- Note: $d_{r,i}$ is a spacing policy that specifies the desired spacing
- This particular controller is nominally stable: i.e., stable if perfectly implemented.
- Spacing policy is easier to design, since it is purely kinematic (运动学的).

Spacing error

- Spacing error

$$e_i(t) = d_i(t) - d_{r,i}(t)$$
$$= (q_{i-1}(t) - q_i(t) - L_i) - \left(r_i + hv_i(t)\right)$$

- q_i = rear-bumper（后保险杠） spacing of vehicle i
- L_i = length of vehicle i



- Control objective: $\lim_{t\to\infty} e_i(t) = 0$ for all i

State-space model

- For each vehicle i

$$\begin{bmatrix} \dot{d}_i \\ \dot{v}_i \\ \dot{a}_i \end{bmatrix} = \begin{bmatrix} v_{i-1} - v_i \\ a_i \\ -\frac{1}{\tau} a_i + \frac{1}{\tau} u_i \end{bmatrix}$$

- d_i = inter-vehicle spacing
- v_i = vehicle speed; v_{i-1} = leading vehicle speed
- a_i = vehicle acceleration
- u_i = control input (desired acceleration)
- τ = time constant associated with driveline (传动) dynamics

A sophisticated controller

- Ploeg et al. proposed a control law for u_i such that

$$h\dot{u}_i = -u_i + \begin{bmatrix} k_p & k_d & k_{dd} \end{bmatrix} \begin{bmatrix} e_i \\ \dot{e}_i \\ \ddot{e}_i \end{bmatrix} + u_{i-1}$$

- Sophistication 1: u_i is now not only the control input, but also an auxiliary state to the state-space model.
- Sophistication 2: u_i depends on not only the state for vehicle i but also the control to vehicle $i-1$.
- Sophistication 3: u_i is not an explicit function of the state, but the solution to an ODE.
- Subscripts for coefficients k: p = proportional, d = derivative, dd = second derivative.

A sophisticated controller

- Ploeg et al. proposed a control law for u_i such that

$$h\dot{u}_i = -u_i + \left[{\color{red}k_p\ k_d\ k_{dd}}\right]\begin{bmatrix} e_i \\ \dot{e}_i \\ \ddot{e}_i \end{bmatrix} + u_{i-1}$$

- Dynamic equation for feedback-controlled system:

$$\begin{pmatrix} \dot{e}_i \\ \dot{v}_i \\ \dot{a}_i \\ \dot{u}_i \end{pmatrix} = \begin{pmatrix} 0 & -1 & -h & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & -\frac{1}{\tau} & \frac{1}{\tau} \\ \frac{k_p}{h} & -\frac{k_d}{h} & -k_d - \frac{k_{dd}(\tau-h)}{h\tau} & -\frac{k_{dd}h+\tau}{h\tau} \end{pmatrix} \begin{pmatrix} e_i \\ v_i \\ a_i \\ u_i \end{pmatrix}$$

$$+ \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & \frac{k_d}{h} & \frac{k_{dd}}{h} & \frac{1}{h} \end{pmatrix} \begin{pmatrix} e_{i-1} \\ v_{i-1} \\ a_{i-1} \\ u_{i-1} \end{pmatrix} \qquad (7)$$

Vehicle model

- The dynamic equation can be compactly written as
$$\dot{x}_i = A_0 x_i + A_1 x_{i-1}$$
- $x_i = [e_i \; v_i \; a_i \; u_i]^T$
- A_0 and A_1 defined accordingly
- For vehicle 1, it follows a virtual reference vehicle 0 such that
$$x_0 = \begin{bmatrix} e_0 \\ v_0 \\ a_0 \\ u_0 \end{bmatrix} = \begin{bmatrix} 0 \\ \bar{v}_0 \\ 0 \\ 0 \end{bmatrix}$$
- $\bar{v}_0$ is the target & equilibrium speed of the platoon

[Not required] String stability

- Recall: platoon is asymptotically stable if spacing error approaches 0, i.e., $\lim_{t\to\infty} e_i(t) = 0 \ \ \forall i$.
- Asymptotic stability: with constant reference speed & no disturbances, platoon will gradually converge to equilibrium state.
- What if there is disturbances?
- String stability: disturbances are not amplified as they propagate in the platoon.
- Ploeg et al. proved that if the controller satisfies

$$k_p > 0, k_d > 0, k_{dd} > 1, (1 + k_{dd})k_d > k_p\tau,$$

then the platoon is string (and thus asymptotically) stable.

[Not required] Frequency domain (SISO)

- How do we know whether disturbances are amplified or weakened in a dynamical system?

Frequency domain response...

- Consider a linear time-invariant (LTI) single-input-single-output (SISO) system

$$\dot{x} = ax + bu$$
$$y = cx + du$$

- Suppose the input is sinusoidal

$$u(t) = \bar{u}e^{j\omega t}$$

- For LTI SISO systems, the output is also sinusoidal

$$y(t) = \bar{y}e^{j\omega t + \phi}$$

- Amplification is characterized by $\frac{\bar{y}}{\bar{u}}$

[Not required] Frequency domain (SISO)

- $\frac{\bar{y}}{\bar{u}}$ determines whether disturbances are amplified or suppressed
  - If $\frac{\bar{y}}{\bar{u}} > 1$, disturbances are amplified, and the system will blow up.
  - If $\frac{\bar{y}}{\bar{u}} < 1$, disturbances are damped, and the system will converge.
- Mathematically, $\frac{\bar{y}}{\bar{u}}$ is equal to the magnitude of the system's frequency response function (FRF) $\Gamma(j\omega)$; hence

$$\frac{\bar{y}}{\bar{u}} \leq \sup_{\omega>0} |\Gamma(j\omega)|.$$

- What is FRF? How to obtain it?

[Not required] Transfer function (SISO)

- Consider a SISO LTI system
$$\dot{x} = ax + bu.$$
- Rearranging leads to $\dot{x} - ax = bu$.
- Let's apply Laplace transform to both sides.
$$\dot{x} - ax \rightarrow sX(s) - aX(s)$$
$$bu \rightarrow bU(s)$$
- Recall: For a signal $f(t)$ defined for $t \in \mathbb{R}_{\geq 0}$, its Laplace transform is given by
$$F(s) = \int_{t=0_-}^{\infty} f(t)e^{-st}dt\,.$$
- Note that $F(s)$ is a function of s.

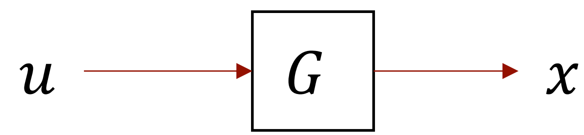
[Not required] Frequency response function (SISO)

- Thus, we have

$$sX(s) - aX(s) = bU(s) \Longrightarrow \frac{X(s)}{U(s)} = \frac{b}{s-a}.$$

- The function $G_{ux}(s) = \frac{b}{s-a}$ is called the transfer function from u to x.
- Replacing s with $j\omega$, where $j = \sqrt{-1}$, we have

$$G_{ux}(j\omega) = \frac{b}{j\omega - a}.$$



- This is the FRF from u to x.
- That is, if the amplitude of u is $\bar{u}$ (e.g., $u(t) = \bar{u}\sin\omega t$), then the amplitude of x is

$$\bar{x}/\bar{u} = \left|\frac{b}{j\omega - a}\right| = \frac{b}{\sqrt{\omega^2 + a^2}}.$$

[Not required] String stability

- Now let's go back to vehicle platooning
- Each vehicle can be modeled by a transfer function $\Gamma_i(j\omega)$ from u_{i-1} to u_i:
$$U_i = \Gamma(s)U_{i-1}.$$
- Then, vehicle i damps disturbance if
$$\sup_{\omega>0}|\Gamma_i(j\omega)| < 1$$
- Therefore, any disturbance will get damped as it propagates through the platoon if
$$\sup_{\omega>0}|\Gamma_i(j\omega)| < 1 \quad \text{for all } i.$$
- The above property is called string stability.

[Not required] Transfer function (no delay)

- Assume no communication delay.
- Recall the proposed controller

$$h\dot{u}_i = -u_i + \begin{bmatrix} k_p \; k_d \; k_{dd} \end{bmatrix} \begin{bmatrix} e_i \\ \dot{e}_i \\ \ddot{e}_i \end{bmatrix} + u_{i-1}$$

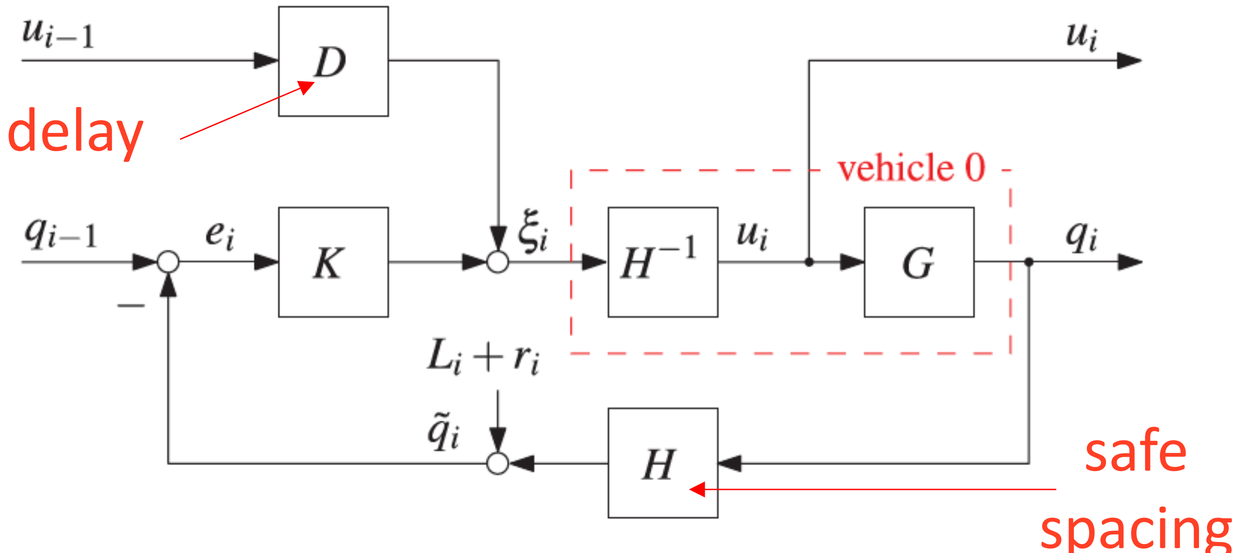
- Setting $e_i, \dot{e}_i, \ddot{e}_i$ to zero, we have

$$h\dot{u}_i = -u_i + u_{i-1}.$$

- Laplace transform: $hsU_i = -U_i + U_{i-1}$.
- So the transfer function from u_{i-1} to u_i is $\Gamma(s) = \frac{1}{hs+1}$.
- Thus, the FRF is $\Gamma(j\omega) = \frac{1}{h(j\omega)+1}$; magnitude always less than 1 -> always string stable!

[Not required] Transfer function (with delay)

- Assume a communication delay of θ between vehicles.
- For the platooning problem, we can obtain the transfer function from the block scheme:



- D, K, H, G are transfer function blocks.
- Delay can be captured by $D(s) = e^{-\theta s}$.

[Not required] Transfer function (with delay)

- Now, consider the platooning problem

$$\begin{bmatrix}\dot{q}_i\\ \dot{v}_i\\ \dot{a}_i\end{bmatrix} = \begin{bmatrix} v_i \\ a_i \\ -\frac{1}{\tau}a_i + \frac{1}{\tau}u_i \end{bmatrix}.$$

- Recall that q_i = absolute position of vehicle i.
- Hence, we have

$$\dddot{q}_i = \dot{a}_i = -\frac{1}{\tau}\ddot{q}_i + \frac{1}{\tau}u_i.$$

- Laplace transform

$$s^3 Q_i(s) = -\frac{s^2}{\tau}Q_i(s) + \frac{1}{\tau}U_i(s).$$

[Not required] Transfer function (with delay)

- Transfer function from U_i to Q_i:

$$G(s) = \frac{1}{s^2(\tau s + 1)}.$$

- Recall that the control input u_i is specified by

$$h\dot{u}_i = -u_i + \begin{bmatrix} k_p & k_d & k_{dd} \end{bmatrix} \begin{bmatrix} e_i \\ \dot{e}_i \\ \ddot{e}_i \end{bmatrix} + u_{i-1}.$$

- Laplace transform:

$$hsU_i = -U_i + \left(k_p + k_d s + k_{dd} s^2\right) E_i + U_{i-1}.$$

- Other transfer functions:

$$K(s) = k_p + k_d s + k_{dd} s^2,$$
$$H(s) = hs + 1.$$

[Not required] Frequency domain

- Equating the input and output to the H^{-1} block leads to

$$U_i = \frac{1}{H(s)}\big(D(s)U_{i-1} + K(s)(Q_{i-1} - G(s)H(s)U_i)\big)$$
$$= \frac{1}{H(s)}\big(D(s)U_{i-1} + K(s)(G(s)U_{i-1} - G(s)H(s)U_i)\big).$$

- Hence, we have the transfer function from u_{i-1} to u_i:

$$\Gamma(s) = \frac{1}{H(s)}\frac{K(s)G(s) + D(s)}{1 + K(s)G(s)}.$$

- Finally, one can show that $\sup_\omega |\Gamma(j\omega)| \le 1$ if

$$k_p > 0, k_d > 0, k_{dd} > 1, (1 + k_{dd})k_d > k_p\tau.$$

- This essentially proves string stability.

Summary

- Technological basis
  - Connected and autonomous vehicles
  - Vehicle platooning
- Simplified formulation
  - Modeling
  - Decision making
  - Final project option 1
- State-of-the-art formulation
  - Modeling
  - Decision making
  - String stability

Why are we doing all this math?

To make their lives, and thus our own lives, easier...