Driver assistance system design A

Adaptive Cruise Control

Carlo Novara

Politecnico di Torino
Dip. Elettronica e Telecomunicazioni

Outline

Introduction

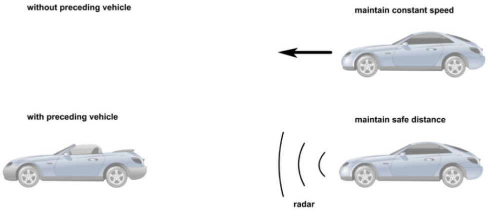
- Adaptive Cruise Control (ACC) is an evolution of standard cruise control, based on a radar or other sensors that measure the distance from the preceding vehicle (PV).
- The goal of ACC is to ensure all the vehicles in the same group (string or platoon) to move at a consensual speed while maintaining the desired spaces between adjacent vehicles.
- Advantages with respect to standard cruise control and "manual" driving:
  - increased traffic capacity,
  - improved safety,
  - improved comfort,
  - reduced fuel consumption.
- Stability properties of the platoon system are fundamental to all the above control objectives.

Introduction

- Classification of possible platoon control approaches:
  - Centralized: A "big" controller receives the required information from all the vehicles, and controls all of them in a coordinated way.
  - Decentralized: Each vehicle controls itself using a reduced quantity of information (e.g., information about itself and the preceding vehicle).
- Centralized control can be impractical, due to the large amount of information that needs to be communicated and also to the large quantity of calculations that must be performed online.
- Another classification is the following:
  - Non-connected/autonomous: Based solely on on-board sensors.
  - Connected/cooperative: Exchange of information with the other vehicles via vehicle-to-vehicle (V2V) communication or with road infrastructure via vehicle-to-infrastructure (V2I) communication.
- In this lecture, we focus on decentralized non-connected ACC systems, based on a leader-follower approach. These are indeed the simplest and most common ACC systems.

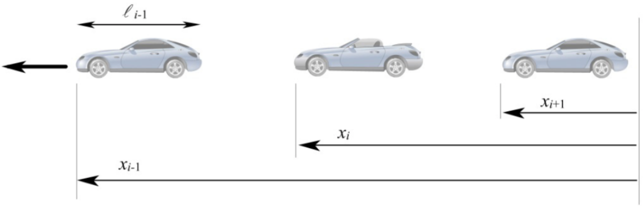
Introduction

- ACC working modes:
  - Standard cruise control: if there are no PVs or the desired speed is lower than the one of the PV.
  - Following mode: longitudinal control (using throttle and brakes) finalized to maintain a given distance (or time-gap) from the PV.

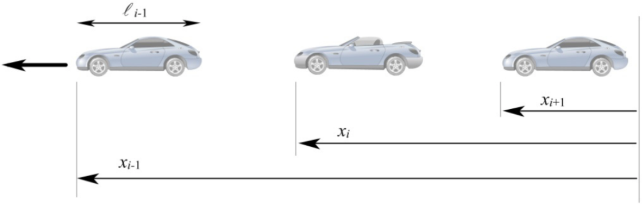


Introduction

- The architecture of an ACC system is hierarchical:
  - Upper level controller: determines the desired vehicle acceleration.
  - Lower level controller: provides the throttle/brake inputs required to have the desired acceleration.
- We focus on the upper controller design (more interesting), supposing that the vehicle is equipped with an effective lower controller.
- We consider a string (platoon) composed of a certain number of vehicles, equipped with a lower controller.



Introduction

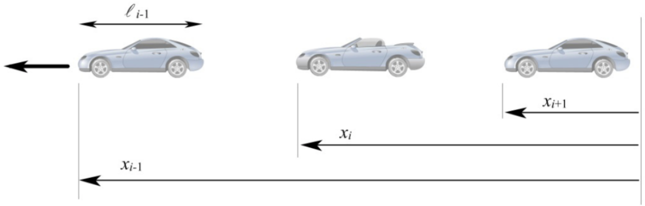


- The longitudinal dynamics of the ith vehicle (including the lower controller) can be approximated by a simple LTI model:

$$\ddot{x}_i = \frac{1}{\tau s + 1} a_i \tag{1}$$

  - x_i: vehicle longitudinal position in an inertial reference frame
  - a_i: commanded acceleration (model input)
  - τ: time constant.

Individual stability



- Important quantities:
  - x_i: vehicle longitudinal position in an inertial reference frame
  - L_{des}: desired space between vehicles
  - $\delta_i \doteq x_i - x_{i-1} + L_{des}$: spacing error.

Definition

An ACC control law is said to ensure individual stability for the ith vehicle if

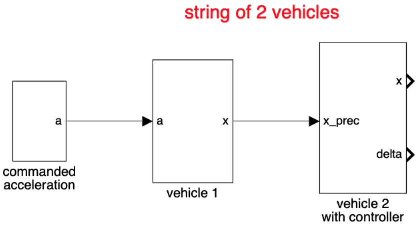
$$\ddot{x}_{i-1} \to 0 \quad \Rightarrow \quad \delta_i \to 0.$$

Example: string with 2 vehicles

- A string with 2 vehicles has been considered. Each vehicle is described by model (1), with $\tau = 0.5\,\mathrm{s}$.
- The first vehicle (leader) of the string is not controlled. The second one (follower) is controlled by a PIDF controller $C(z)$ designed using the Simulink tuner (transfer function based method):

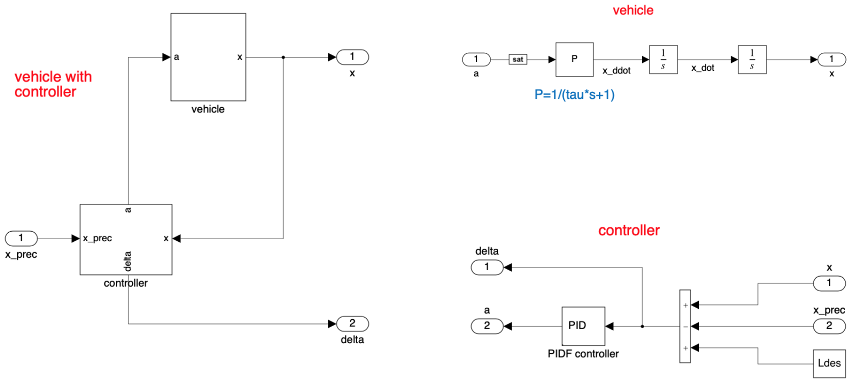
$$a_i = C(z)\delta_i.$$

- The string was implemented in Simulink.



Example: string with 2 vehicles

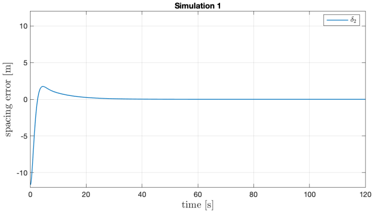
- The block "vehicle with controller" is here expanded.



Note: The controller needs to measure only the relative distance $x_i - x_{i-1}$.

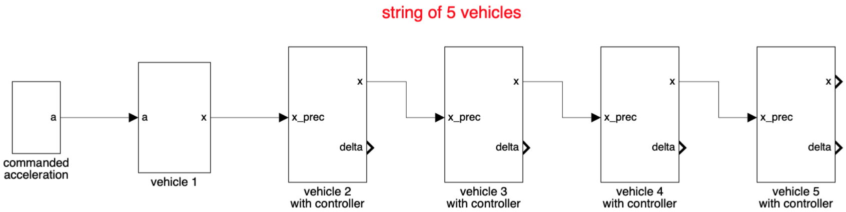
Example: string with 2 vehicles

- Two simulations were carried out:
  - Vehicle initial positions: chosen to have initial random spacing errors.
  - Initial speeds: $\dot{x}_i(0) = 80\,\mathrm{km/h}$ for all vehicles.
  - $L_{des} = 60\,\mathrm{m}$.
  - Commanded acceleration: $a_1 = \begin{cases} 0 & \text{simulation 1} \\ \sin(0.3t) & \text{simulation 2.} \end{cases}$
- As expected, in simulation 1, $\delta_2 \to 0$; in simulation 2, $\ddot{x}_1$ changes and δ_2 remains bounded, confirming individual stability.



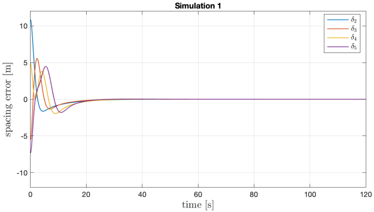
Example: string with 5 vehicles

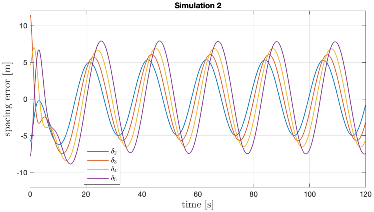
- A string with 5 vehicles has been considered. Each vehicle is described by model (1).
- The leader is not controlled. The followers are controlled by the same $C(z)$ developed above.
- The string was implemented in Simulink. The blocks “vehicle i with controller” are all equal to the one described above.



Example: string with 5 vehicles

- Two simulations were carried out under the above setting.
- In simulation 1, as expected, $\delta_i \to 0, \ \forall i$.
- In simulation 2, the spacing error δ_i increases with i, namely, with the number of vehicles in the platoon. This phenomenon is called string instability.





Individual stability

- During acceleration or deceleration of the preceding vehicle, the spacing error may be non-zero.
- **Problem:** Undesired propagations of the spacing error from the preceding vehicles to subsequent vehicles in a string may yield string instability, even if each vehicle is individually stable.
- String instability is an issue by which the spacing errors of subsequent vehicles are larger than those of preceding vehicles.
- String stability is the property by which all the spacing errors in a string remain “small”.
- A formal definition of string stability will be given below, together with suitable stability conditions.
- To this aim, we need to introduce basic notions about vector, signal and system norms.

Norms

Definition

Consider a linear vector space F over the field $\mathbb{C}$. A **norm** on F is a function $\|\cdot\| : F \to \mathbb{R}$ such that, for all $f, g \in F$ and for all $a \in \mathbb{C}$, the following properties are satisfied:

1. Non-negativity: $\|f\| \geq 0$.
2. Positive homogeneity: $\|af\| = |a| \, \|f\|$.
3. Triangular inequality: $\|f + g\| \leq \|f\| + \|g\|$.
4. Point-separating: $\|f\| = 0$ if and if $f = 0$.

- **Important meanings:**
  - $\|f\|$: length (or magnitude) of a vector f.
  - $\|f - g\|$: distance between two vectors f and g.

Vector norms

- Consider a vector $f = [f_1; f_2; \ldots; f_n] \in \mathbb{R}^{n\times 1}$.

$$\begin{aligned}
\ell_2 \text{ norm}: \quad & \|f\|_2 \doteq \sqrt{\sum_{i=1}^{n} f_i^2} = \sqrt{f^\top f} \\
\ell_1 \text{ norm}: \quad & \|f\|_1 \doteq \sum_{i=1}^{n} |f_i| \\
\ell_\infty \text{ norm}: \quad & \|f\|_\infty \doteq \max_{i=1,\ldots,n} |f_i| \\
\ell_2 \text{ weighted norm}: \quad & \|f\|_{Q,2} \doteq \sqrt{\sum_{i=1}^{n} q_i f_i^2} = \sqrt{f^\top Q f}
\end{aligned}$$

where $Q = \mathrm{diag}(q_1, \ldots, q_n)$, $q_i \geq 0$.

Vector norm computation

Consider the vector $f = [1 \quad -2 \quad 3]^T \in \mathbb{R}^3$.

Matlab

- ℓ_2 norm: $\|f\|_2 = \sqrt{|1|^2 + |-2|^2 + |3|^2} = \sqrt{14} = 3.7417$ `norm(f)=norm(f,2)=3.7417`

- ℓ_1 norm: $\|f\|_1 = \sum_{i=1}^{n} |f_i| = |1| + |-2| + |3| = 6$ `norm(f,1)=6`

- ℓ_∞ norm: $\|f\|_\infty = \max_{i=1,\ldots,n} \{|1|, |-2|, |3|\} = 3$ `norm(f,inf)=3`

Signal norms

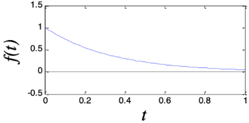
- A signal is a function of time: $f \equiv f(t)$, where $t \in \mathbb{R}^+ \doteq [0, \infty]$.
- Consider a signal $f : \mathbb{R}^+ \to \mathbb{R}$.

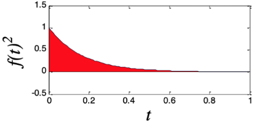
$$
\begin{aligned}
L_2 \text{ norm :} \quad & \|f\|_2 \doteq \sqrt{\int_0^\infty f(t)^2 dt} \\
L_1 \text{ norm :} \quad & \|f\|_1 \doteq \int_0^\infty |f(t)|\, dt \\
L_\infty \text{ norm :} \quad & \|f\|_\infty \doteq \sup_{t \in \mathbb{R}^+} |f(t)|\,.
\end{aligned}
$$

- **Interpretation:**
  - $\|f\|_2^2$: (generalized) energy of the signal f.
  - $\|f\|_\infty$: amplitude of the signal f.

Signal norm computation

Consider the function $f(t) = e^{-3t}$





$$\|f\|_2^2 = \int_0^\infty f(t)^2\,dt = \int_0^\infty e^{-6t}\,dt = \left[-\frac{1}{6}e^{-6t}\right]_0^\infty = \frac{1}{6}$$

$$\|f\|_\infty = \max_{t\in R^+}|f(t)| = \max_{t\in R^+}\left|e^{-3t}\right| = 1$$

Matlab

```
dt=0.0001;
t=0:dt:10;
f=exp(-3*t)';

norm(f)*sqrt(dt)            % L2 norm

norm(f,1)*dt                % L1 norm

norm(f,inf)                 % Linf norm
```

For vector-valued functions, norm computation can be slightly more complicated.

System norms

- Consider linear time invariant (LTI) single-input-single-output (SISO) system described by a transfer function $G(s)$ whose poles have all negative real part. $s = \sigma + j\omega$ is the Laplace variable.
- Common system norms are the following:

$$\begin{aligned} H_2 \text{ norm}: & \quad \|G\|_2 \doteq \sqrt{\frac{1}{2\pi}\int_{-\infty}^{\infty} |G(j\omega)|^2 \, d\omega} \\ H_\infty \text{ norm}: & \quad \|G\|_\infty \doteq \sup_{\omega \in \mathbb{R}} |G(j\omega)| \\ L_1 \text{ norm}: & \quad \|g\|_1 \doteq \int_0^{\infty} |g(t)| \, dt \end{aligned}$$

where $g(t)$ is the inverse Laplace transform of $G(s)$:

$$g(t) \doteq \mathscr{L}^{-1}\{G(s)\}$$

(it can be shown that $g(t)$ is the impulse response of $G(s)$).

System norms

- Let u be a generic input signal of $G(s)$, and y the corresponding output signal. Let $\|u\|_*$ and $\|y\|_*$ their signal norms.
- **Properties:**
  - $\|G\|_\infty$ is the input-output L_2 gain: $\|y\|_2 \leq \|G\|_\infty \|u\|_2$, $\forall u$.
  - $\|g\|_1$ is the input-output L_∞ gain: $\|y\|_\infty \leq \|g\|_1 \|u\|_\infty$, $\forall u$.
  - $\|G\|_2$ is the energy-to-amplitude gain: $\|y\|_\infty \leq \|G\|_2 \|u\|_2$, $\forall u$.
- They are also called *induced norms*.

System norm computation

System:

```
G=zpk(-1,[-3+1i -3-1i],1)
```

```
G =

        (s+1)
   ---------------
   (s^2 + 6s + 10)

Continuous-time zero/pole/gain model.
```

Stability verification:

```
pole(G)
```

```
ans = 2×1 complex
   -3.0000 + 1.0000i
   -3.0000 - 1.0000i
```

H_2 **norm:**

```
norm(G)
```

```
ans = 0.3028
```

```
norm(G,2)
```

```
ans = 0.3028
```

H_∞ **norm:**

```
norm(G,inf)
```

```
ans = 0.1756
```

Alternative:

```
hinfnorm(G)
```

```
ans = 0.1756
```

L_1 **norm:**

```
dt=0.01; % chosen sufficiently small
T=50;    % chosen sufficiently large
g=impulse(G,0:dt:T);
norm(g,1)*dt
```

```
ans = 0.2658
```

If the system is unstable:

norm(G,2) gives the L_2 norm.

norm(G,inf) gives the L_∞ norm.

hinfnorm(G) = ∞.

norm(g,1) $\to \infty$ as $T \to \infty$.

String stability

- String stability is the property by which all the spacing errors in a string remain "small". A definition of string stability is the following.

Definition

A string is stable if $$\|\delta_i\|_\infty \leq \|\delta_{i-1}\|_\infty\,, \quad \forall i > 1. \tag{2}$$

- Motivation: condition (2) ensures that the spacing errors from the preceding vehicles to the subsequent vehicles do not increase.
- Recalling the above induced norms, (2) is satisfied if

$$\|g\|_1 \leq 1 \tag{3}$$

where $g \equiv g(t)$ is the impulse response of the transfer function $G(s) \doteq \delta_i(s)/\delta_{i-1}(s)$ (tf from δ_{i-1} to δ_i).

String stability

- Inequality (3) is thus a condition for string stability. However, designing controllers that satisfy (3) is in general difficult.
- A result taken from LTI system theory can be used, asserting that condition (3) is satisfied if $\|G\|_\infty \leq 1$ and $g(t)$ does not change sign.
- Designing a controller to satisfy $\|G\|_\infty \leq 1$ is simpler. The condition on $g(t)$ sign can be checked after the design.

Theorem (see Rajamani's book)

A string is stable if both the following conditions are satisfied:

(i) $\|G(s)\|_\infty \leq 1$.

(ii) $g(t)$ *has the same sign* $\forall t \geq 0$.

- Several string stability concepts can be found in the literature. Some of them just require condition (i) and are thus weaker than the one presented here.

Control policies

- Consider a string composed of a certain number of vehicles, equipped with the same lower level controller.
- A simple model of the longitudinal dynamics of the ith vehicle (including the lower controller) in the string is

$$\ddot{x}_i = \frac{1}{\tau s + 1} a_i \tag{4}$$

  - x_i: vehicle longitudinal position in an inertial reference frame
  - a_i: commanded acceleration (model input)
  - τ: time constant.
- The goal is to design the upper level controller. Common policies:
  - Constant inter-vehicle spacing policies are not suitable, since usually they not ensure string stability, see the above example.
  - Constant time-gap (CTG) policies proved to work well.

CTG control policy

- Let h be a desired *time-gap* between the vehicles in the string. Then, the desired space between vehicles can be defined as

$$L_{des} \doteq h\dot{x}_i.$$

- The spacing error is given by $\delta_i \doteq \varepsilon_i + L_{des}$, where $\varepsilon_i \doteq x_i - x_{i-1}$.
- An effective CTG control policy is given by the following PD law:

$$a_i = -\frac{1}{h}\left(\lambda\delta_i + \dot{\varepsilon}_i\right) \tag{5}$$

  where λ is a parameter to choose.

- Relatively simple calculations[1] show that the transfer function from δ_{i-1} to δ_i is given by

$$G(s) = \frac{\delta_i(s)}{\delta_{i-1}(s)} = \frac{s+\lambda}{h\tau s^3 + hs^2 + (1+\lambda h)s + \lambda}.$$

[1] Rajamani's book.

CTG control policy

Theorem (see Rajamani's book)

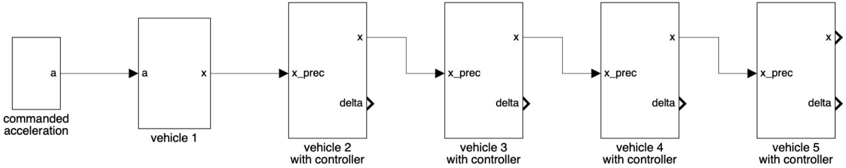
Consider a string of vehicles described by model (4)*. Assume that:*
(i) $h \geq 2\tau$.
(ii) $g(t) = \mathscr{L}^{-1}\{G(s)\}$ *has the same sign* $\forall t \geq 0$.
Then, a value of λ *exists such that the CTG control policy* (5) *ensures string stability.*

- Reducing the time-gap h would allow an increased traffic capacity. However, the theorem shows that h cannot be smaller than 2τ, otherwise string instability may occur.
- A realistic value is $\tau \cong 0.5\,\mathrm{s}$, which implies $h \gtrsim 1\,\mathrm{s}$ (lower bound).
  - This is equivalent to a steady-state spacing of about $30\,\mathrm{m}$ between vehicles at a speed of $30\,\mathrm{m/s}$.
  - The maximum traffic flow rate that can be achieved is therefore less than $3600\,\mathrm{vehicles/hour}$.

Example: string with 5 vehicles

- A string with 5 vehicles has been considered. Each vehicle is described by model (1), where $\tau = 0.5\,\mathrm{s}$ and $a_i \in [-10, 3]\ \mathrm{m/s}^2$.
- The first vehicle of the string is not controlled. The other ones are controlled by the CTG control policy (5).
- The string was implemented in Simulink. The blocks "vehicle i with controller" are all equal to the one described above, except the controller.

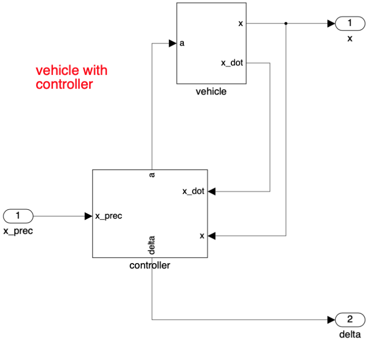
string of 5 vehicles



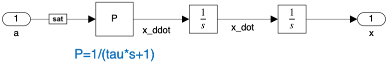
Example: string with 5 vehicles

- The block “vehicle with controller” is here expanded.

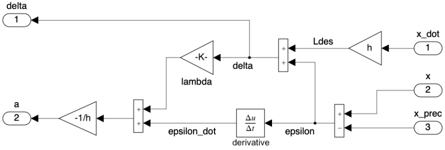
vehicle with controller



vehicle



controller

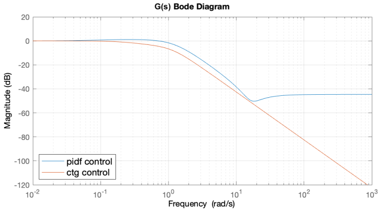


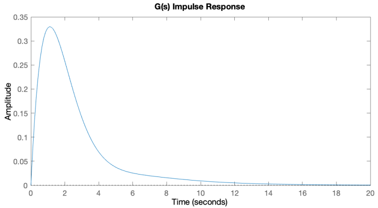
Example: string with 5 vehicles

- The following parameter values of the CTG policy were chosen:
  - time gap $h = 2.7\,\mathrm{s}$; at a speed $v_x = 80\,\mathrm{km/h}$ this h corresponds to a distance of $60\,\mathrm{m}$; note that $h > 2\tau$;
  - $\lambda = 0.5$, chosen by trial-and-error; the choice is not critical.
- With these parameters, the transfer function from δ_{i-1} to δ_i is

$$G(s) = \frac{s + 0.5}{1.35s^3 + 2.7s^2 + 2.35s + 0.5}.$$

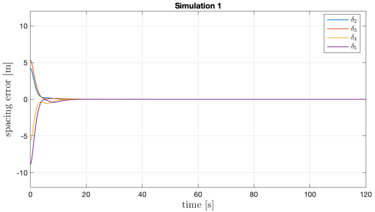
- Using the Matlab commands `hinfnorm` and `impulse`, it can be verified that $\|G(s)\|_\infty = 1$ and $g(t) \geq 0$, $\forall t \geq 0$.

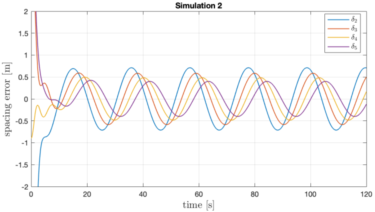




Example: string with 5 vehicles

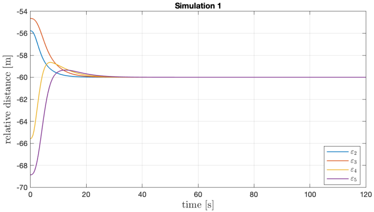
- Two simulations were carried out under the same setting considered in the previous example (with $a_1 = 0$ or $a_1 = \sin(0.3t)$).
- In simulation 1, as expected, $\delta_i \to 0, \ \forall i$.
- In simulation 2, the spacing error δ_i does not increase with the number of vehicles in the platoon, showing string stability.

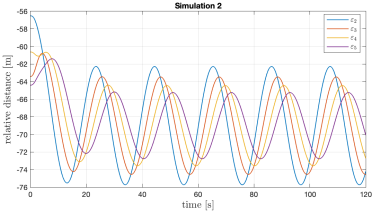




Example: string with 5 vehicles

- In the case of CTG policy, the spacing error δ_i is related to the actual distance between two vehicles ε_i.
- The distance oscillations, shown in the following figures, do not increase with the number of vehicles.





Discussion

- Decentralized non-connected ACC systems, based on a leader-follower approach, have been considered.
- Individual stability is not sufficient to have “small” spacing errors between vehicles in a platoon.
- String stability guarantees a reliable behavior of a platoon of vehicles. An interesting point is that the vehicles are controlled individually but they have a satisfactory collective behavior.
- A relatively simple setting has been considered here. More detailed ones can be found in the literature, involving nonlinear vehicle models and more sophisticated control policies.