# Automotive Control Systems (EE 5812/ MEEM 5812)
## Project #4
## Intelligent Cruise Control Design
**Submit: Your one-page question answer sheet in Canvas as a pdf file, your three Simulink controllers in Canvas as slx-files, and your gain generation script as an m-file.**

An intelligent cruise control system is required to maintain speed when the following distance to any lead vehicles is greater than a safe distance and also to maintain a safe following distance (headway) when the lead vehicle is sufficiently close. Note that we must convert between two control systems: One that maintains speed and one that maintains following distance. The switch from speed control to distance control should occur when the lead vehicle hits a distance threshold. The switch from distance control to speed control should occur when the speed hits a threshold. To minimize transients when switching between these control systems, each system should be initialized appropriately, as discussed in the notes.

I have supplied a vehicle simulation, including disturbance inputs, as a Simulink diagram. This Simulink file appears in the Canvas assignment as a file "IntelligentCruiseControl.slx" The disturbances to this system are a lead vehicle speed (that forces us to change our speed), a road gradient change, and changing commanded speeds. This file also contains a three controller subsystem blocks. Your job will be to populate these controller blocks such that you achieve your objectives. This will be done in pieces: Design a velocity control system and test; Design a headway control system and test; Put these two controllers together with the logic and initialization to yield the full intelligent cruise control system and test.

The linearized plant is modelled:

$$G_p(s) = \frac{2*10^{-4}}{s+0.02}$$

The control input (traction force) to this system saturates at ±5,000. Note that the saturation in the model is ±1e7. This is required for the initial transient, but you should verify that your control magnitude stays less than 5000 (except during this initial transient).

To make the design easier, only use a single integrator in your controller. To avoid people working with previous designs, <u>you will not get credit for the project if you use two integrators in your controller.</u>

1. The first problem is to design a velocity control system for the linearized plant. Here, we will ignore the headway measurement and just follow the speed. The controller should yield a

steady-state error of zero for a step input, a 2% settling time of 15 seconds, and a damping ratio of 0.8. Note that the notes include a description of this design process (but the numbers are different for this problem). Implement this in the controller block and run the simulation. Note that the speed should be controlled to the desired values, but the headway may become negative (indicating you hit the lead car). This headway problem will be fixed in later designs. Verify that the performance of this controller is acceptable by selecting 'Velocity Control' in the selection box that appears when starting the simulation.

The specifications for the velocity control are:
% Overshoot in velocity is less than 3%;
The 2% settling time is less than 15 s;
The steady state error in velocity is zero (actually tested to be less than $10^{-5}$);
The speed difference caused by the gradient change is less than 0.5 m/s;
The traction force stays less than 5000 N (except for the initial unrealistic transient).

2. The second part is to design a headway control system for the linearized plant. The headway should conform to the 3 second rule, i. e., the headway should be maintained to yield a trailing distance that is three times the desired speed. Note that the headway should be 3 times the actual speed, but it was easier for me to program it to be 3 times the commanded speed. This will be close to the actual speed and will be safe because it will yield more than three times the actual speed. This controller should be a PIID (a PID with an extra integral as described in the notes). Use LQR (with the two integrals) to design this controller as discussed in the notes. Implement this headway controller in the controller block and run the simulation. Verify that the performance of this controller is acceptable by selecting 'Headway Control' in the selection box that appears when the starting the simulation.

The specifications for the headway control are:
The maximum absolute value of the headway error (after the initial transient, i. e., after 100 seconds) is less than 1 m;
The reaction time of the headway controller is less than 4 s. This specification is measured with respect to a ramp in headway and is atypical, but still provides info on how fast the controller reacts;
The maximum tractive force (ignoring the initial transient) is less than 5000 N.

3. Now the hard part. Combine the headway and the speed controllers such that speed is maintained when the lead vehicle is far away and the headway is maintained when the vehicle is

close. This is accomplished by switching between the two controllers. When switching controllers, you want bumpless transfer, i. e., the control after the switch should be the same as the control before the switch (this avoids the large transients that result from the integrators accumulating the error when the controller is not in use). Bumpless transfer is accomplished by resetting the integral values to the proper values when switching between controllers. Verify that the performance of the total controller is acceptable by selecting 'Intelligent Control' in the selection box that appears when the starting the simulation.

The specification for the total control are:
The maximum absolute value of the headway error (when headway control is active from 160 to 260 s) is less than 2 m;
The steady-state error of velocity control is less than 0.001 m/s;
The minimum headway (between 100 and 600 seconds in the simulation) is more than 68 m;
The traction force (between 100 and 600 seconds in the simulation) is more than -5000 N.

Upload your three controllers as Simulink files (slx-files) in Canvas. You should also upload a single Matlab file (m-file) for computing the controller gains (these can be hard coded into this file if you do the design without Matlab) and for computing the Headway Control gains generated via LQR theory. I suggest that you use variables in your controller for these gains. This makes changing cost function parameters simple and avoids the problem of mistyped gains.

Provide short answers, (a few sentences each) on a single page, to the following questions (and upload this document to canvas as a pdf-file):

1. What happens when the velocity command is ramped down (driver pushed the decelerate button) at 100 seconds into the simulation? Let me know what happens to the tractive force and the velocity. Note that you should be operating on velocity control at this point.

2. What happens when the vehicle starts going up a hill at 500 seconds into the simulation? Let me know what happens to the tractive force and the velocity. Note that you should be operating on velocity control at this point.

3. What happens when the lead vehicle starts accelerating at 250 seconds into the simulation? Let me know what happens to the tractive force, velocity, headway. Note that you should be operating on headway control at this point.

4. What happens when you switch from velocity control to headway control at around 170 seconds into the simulation? Let me know what happens to the tractive force, velocity, headway.

5. What happens when you switch from headway control to velocity control at around 300 seconds into the simulation? Let me know what happens to the tractive force, velocity, headway.

Comments:

Each student should do this project individually and no student should share Simulink diagrams or code or results with other students. Note that I will be running your results through a similarity checker. If you have shared results and it is detected, you will fail this course! So, please don't share code. But, feel free to ask your instructor questions or check results with your instructor.