University of California at Riverside, Department of Mechanical Engineering
ME 120, Winter 2017, Prof. Elisa Franco

Matlab Homework Assignment

Homework objectives:

- Learn how to use Simulink (MATLAB)
- Understand proportional-integral control (Weeks 8 and 9)

A Simulink model for a car cruise control system is posted on iLearn. Homework questions:

1. Leaving the control gains (orange block) at their default values, plot the response of the system to a step input in the speed reference value: at time $t = 0$ we go from an initial reference speed of 40 mi/h to a final speed 50 mi/h. Measure the time it takes for the system error to settle within 1% of the desired final speed (50 mi/h).
2. Tune the control gains (orange block) and design a controller that settles at least 50% faster than the default controller provided. Include the gains you used, a plot of the closed loop response, and describe any undesireable features in the solution you obtain.
3. Now add a hill (cyan block) of 30% slope (0.3) at time $t = 60$ and comment on the performance of your controller in rejecting this disturbance.

Your homework must be a typeset report which includes a printout of the Simulink model, plots with labeled axes, having reasonable axis limits, captions and well written response to both questions. A template for how your report should look like is uploaded on iLearn.

Description of the system

We will refer to the cruise control system discussed in Chapter 3, Section 3.1 of the Åstrom-Murray textbook [1]. The goal of a cruise controller is to maintain a constant desired speed despite disturbances; we will consider changes of road slope as our disturbance. As shown in Figure , the control system is composed of sensors (detect the actual speed), a controller, and actuators (gas injection system). Depending on the error between the desired and the measured (actual) speed, the controller decides how much gas to inject in the engine. If you do not have a cruise control system in your car, **you** are the controller (sensors are your eyes on the speedometer, and actuator is your foot on the accelerator). If you do have a cruise controller, then sensors and actuators are circuits suitably designed to do the job for you. We will learn what should be the *transfer function* of the controller, and not worry about the actual synthesis with circuit elements.

For example, if the road slope increases, the controller will have to inject more gas in the car engine, to increase the power provided to the car; if the slope decreases, the controller will decrease the gas amount injected in the engine. Because the response of the controller is opposite relative to the sign of the error, we call this *negative feedback*.

Before we look at the controller, we need to derive a model for the system we want to control: the car.

1. **Dynamics of the vehicle.** We use Newton's law. F is the force generated by the wheels on the road; F_d comprises forces due to gravity, friction and aerodynamic drag. If a is the vehicle acceleration and m is its mass:

$$ma = F - F_d$$

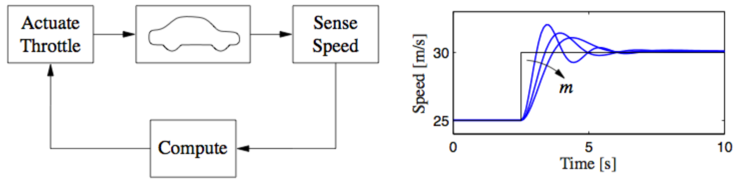

Figure 1: Scheme: proportional-integral speed control for a vehicle (figure taken from [1]); the vehicle mass varies between 1000 and 3000 kg. Notice that while the transient changes, at steady state we always reach the desired speed for very different values of the vehicle mass!

2. **Input: Force generated by the engine.** The vehicle speed is depends on the amount of gas injected in the engine. The amount of gas is controlled by the throttle position, which we take as our control input and we will call it u: we will assume that $0 \leq u \leq 1$ ($u = 0$ means no gas; $u = 1$ means full throttle).

$$F = u\, \alpha_n\, T(\alpha_n\, v),$$

Where:

- u: control input
- $\alpha_n = \frac{n}{r}$, where n is the gear ratio and r is the wheel radius. Typical values for α_n at different gears are: $\alpha_1 = 40, \quad \alpha_2 = 25, \quad \alpha_3 = 16, \quad \alpha_4 = 12, \quad \alpha_5 = 10$
- $T(\alpha_n\, v)$ is the torque of the engine as a function of the gear chosen and the velocity of the engine. The torque curves are nonlinear, see Figure 2. Typically, the torque curves have this form:

$$T(\omega) = T_m \left(1 - \beta \left(\frac{\omega}{\omega_m} - 1\right)^2\right), \qquad \omega = \alpha_n v \quad \text{is the angular velocity of the engine}$$

Typical values: $T_m = 190Nm$; $\omega_m = 420$ rad/s, about 4000 RPM; $\beta = 0.4$.

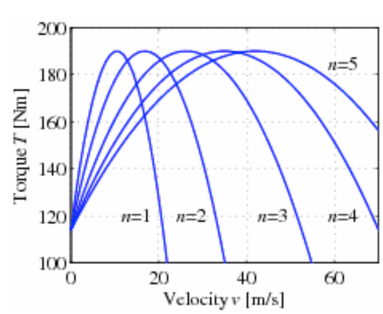

Figure 2: Torque as a function of car speed for different gears. (Figure taken from [1])

3. **Disturbance force** F_d. The disturbance has three components:

   (a) F_g, force due to gravity. This is the disturbance that we will vary in our model.

   $$F_g = mg \sin\theta,$$

   where g is gravity, m is the mass of the vehicle (between 1000-3000kg), and θ is the slope angle of the road.

(b) F_r, force due to rolling friction. We will assume

$$F_r = mgC_r sign(v),$$

where C_r is the coefficient of rolling friction, which we will take as $C_r = 0.01$, and $sign(v)$ is the sign of the vehicle speed ($sign(v)$=0 if $v = 0$).

(c) F_a is the aerodynamic drag. We will take:

$$F_a = \frac{1}{2}\rho C_d A v^2,$$

where $\rho = 1.3k/m^3$ is the air density, $C_d = 0.32$ is the shape-dependent drag coefficient and $A = 2.4m^2$ is the frontal area of the car.

Matlab files

The Simulink model for the system is uploaded on iLearn. Your job is to understand how to edit it, use it and tune the parameters of the controller to achieve the assigned objectives.

Timeline:

- Week 9: Download the model, familiarize yourself with the system description and with Simulink. Learn how to tune the controller gains.
- Week 10: Plot your results and typeset your solution.
- Printed homework report is due on March 16 (last day of class), AT THE BEGINNING OF CLASS.

References

[1] K. J. Åstrom and R.M. Murray. *Feedback Systems: An Introduction for Scientists and Engineers*. Princeton University Press, 2009.