

HW 1 Cruise Control Model

Richard M. Murray Shreesh Mysore
Control and Dynamical Systems
California Institute of Technology

30 September 2002

The file `hw1cruise.mdl` contains a simple model of a cruise controller for a car, depicted in Figure 1. The system consists of a simple model of a vehicle (“car dynamics”), with a proportional plus integral (PI) control law to track a step change in the car velocity. The commanded and actual variable are stored to the MATLAB workspace, along with the time vector.

The vehicle model is given by a simple linear model

$$m\dot{v} = -bv + F_e$$

where v is the vehicle velocity, m is the mass of the vehicle, b is the damping coefficient and F_e is the force applied by the engine. We assume linear damping ($-bv$). The model is initialized at 55 miles/hour. In addition a 0.2 second time delay is included to simulate the engine dynamics, and the input force is saturated at a value corresponding to a maximum velocity of 100 miles/hour. These dynamics are in the subsystem block labelled “Car dynamics”.

The control law attempts to maintain the vehicle speed by applying a motor force that is the sum of two terms:

- a *proportional gain* that depends on the difference in speed between the commanded input and current vehicle speed
- an *integral gain* that depends on the integrated error of the commanded and current speed.

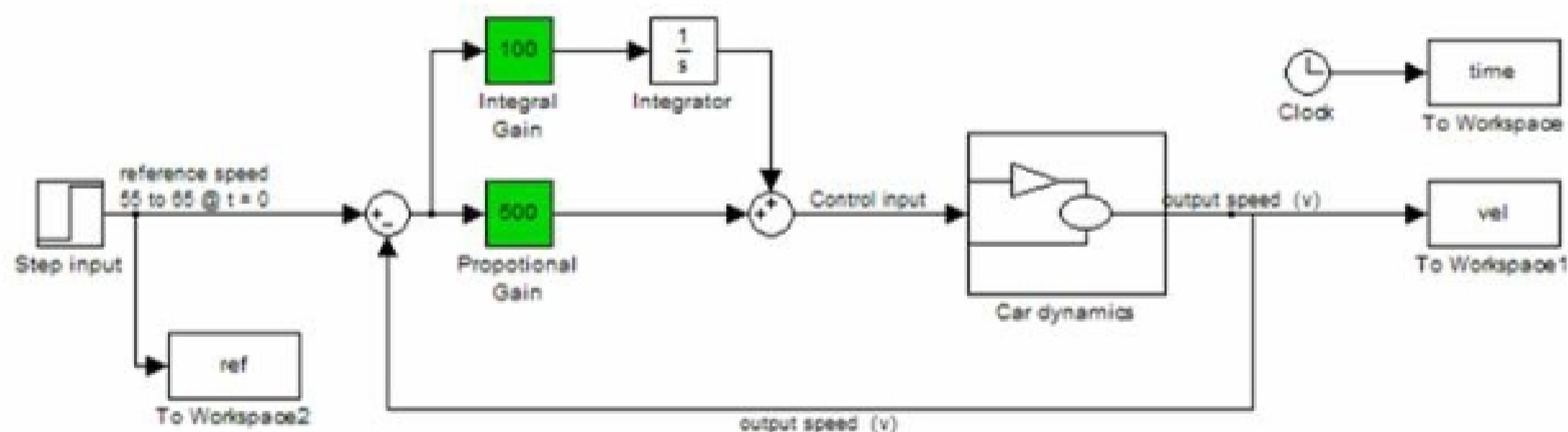


Figure 1: SIMULINK model for `hw1cruise.mdl`.

The commanded speed is set by a step input, which changes from 55 miles/hour to 65 miles/hour at $t = 10$ seconds. The integrator is initialized so that the initial input from the control law keeps the car at 55 miles/hour, the initial speed of the car.

The gains for the control law are shown in the green (shaded) boxes. They can be set by double clicking on them and typing the new value in the dialog box. The default gains give a stable control law with some overshoot and a fairly long settling time.

The simulation store three variables to the MATLAB workspace:

- `ref` – the commanded (reference) velocity for the car
- `vel` – the actual velocity of the car
- `time` – the time vector for the simulaton

These variables are all stored as simple arrays, allowing them to be plotted using the `plot` command.