ES 155 Homework 6

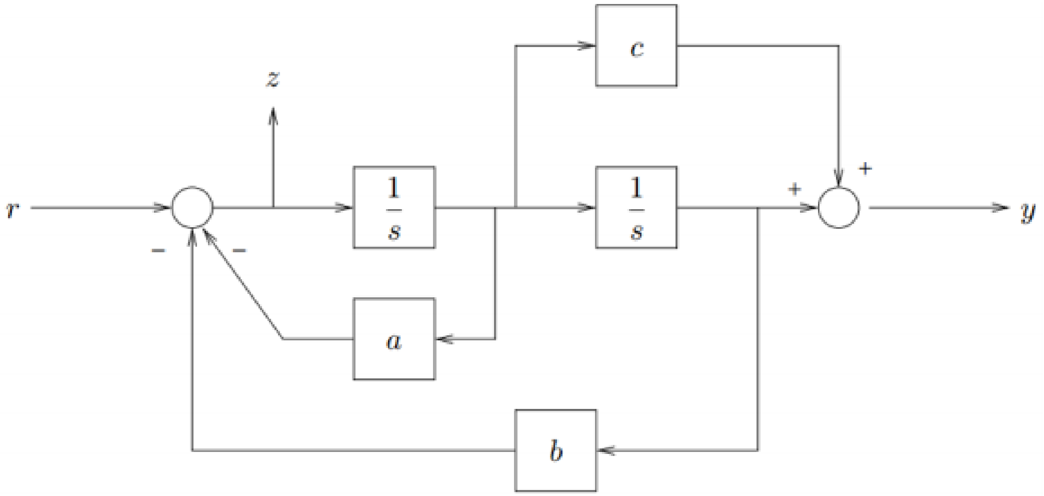

Figure 1: System Block Diagram

1. (a) To compute the transfer function for the block diagram in Figure 1, first add some additional labels. Let Z be the output of the first summing node on the left, A be the output of the first integrator, and B be the output of the second integrator. This gives

$$\begin{aligned} Z(s) &= R(s) - aA(s) - bB(s) \\ Z(s) &= sA(s) \\ A(s) &= sB(s) \\ Y(s) &= B(s) + cA(s) \end{aligned}$$

With four equations and four unknowns, we can plug in to solve for $Y(s) = H_{yr}(s)R(s)$

$$\begin{aligned}
Y(s) &= B(s) + cA(s) = B(s) + csB(s) \\
B(s) &= \frac{Y(s)}{1+cs} \\
Z(s) &= sA(s) = R(s) - aA(s) - bB(s) \\
s^2B(s) &= R(s) - asB(s) - bB(s) \\
R(s) &= \left(s^2 + asB(s) + bB(s)\right) B(s) \\
\frac{Y(s)}{1+cs} &= \frac{1}{s^2+as+b} R(s) \\
H_{ry} &= \frac{1+cs}{s^2+as+b}
\end{aligned}$$

(b) The state space system

$$\begin{aligned}
\dot{x} = Ax + Bu = \frac{d}{dt}\begin{bmatrix} x_1 \\ x_2 \end{bmatrix} &= \begin{bmatrix} 0 & 1 \\ -a_2 & -a_1 \end{bmatrix}\begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} r \\
y = Cx + Du = \begin{bmatrix} b_2 & b_1 \end{bmatrix}\begin{bmatrix} x_1 \\ x_2 \end{bmatrix} &+ d \cdot r
\end{aligned}$$

can be written

$$\begin{aligned}
\dot{x_2} &= -a_2x_1 - a_1x_2 + r \\
&= -a_2x_1 - a_1\dot{x_1} + r \\
y &= b_2x_1 + b_1x_2 + dr \\
&= b_2x_1 + b_1\dot{x_1} + dr
\end{aligned}$$

In the Laplace domain

$$\begin{aligned}
s^2X_1 &= -a_2X_1 - a_1sX_1 + R \\
X_1 &= \frac{1}{s^2+a_1s+a_2} R \\
Y &= b_2X_1 + b_1sX_1 + d \cdot R \\
&= (b_2 + b_1s)\frac{1}{s^2+a_1s+a_2} R + d \cdot R \\
&= \left(\frac{b_2+b_1s}{s^2+a_1s+a_2} + d\right) R
\end{aligned}$$

which is has the same transfer function as part a) for $a_1 = a, a_2 = b, b_1 = c, b_2 = 1, d = 0$.

(c) The transfer function H_{zr} can also be computed.

$$\begin{aligned} Z &= R + aA - bB = R - a\frac{1}{s}Z - b\frac{1}{s^2}Z \\ Z\left(1 + \frac{a}{s} + \frac{b}{s^2}\right) &= R \\ Z &= \frac{s^2}{s^2 + sa + b}R \\ H_{rz} &= \frac{s^2}{s^2 + sa + b} \end{aligned}$$

2. A simple cruise control system has the equations of motion

$$\begin{aligned} m\dot{v} &= -cv + b\tau + F_{\text{hill}} \\ \dot{\tau} &= a(-\tau + Tu) \end{aligned}$$

(a) With controller $u = k_p e$ where $e = r - v$, the system can be written as a block diagram of each of the components. First compute their transfer functions need to be computed. $H_U = k_p$ is easy. The engine torque transfer function is

$$\begin{aligned} s\tau &= -a\tau + aTU \\ \tau &= \frac{aT}{s+a}U \\ H_\tau &= \frac{aT}{s+a} \end{aligned}$$

Finally, the velocity in the s-domain (where F_{hill} is d) can be written

$$\begin{aligned} msV(s) &= -cV(s) + b\tau(s) + D(s) \\ V(s) &= \frac{b\tau(s) + D(s)}{ms + c} \end{aligned}$$

Because it takes two inputs ($\tau(s)$ and $D(s)$), this block is broken up in the block diagram, shown in Figure

(b) The transfer function of the entire system can be broken down as several individual blocks, as shown in part (a). Because the transfer function of two blocks connected in series is the product of their indivdual transfer functions, the overall transfer function can be computed in MATLAB using this technique along with the `feedback` function.

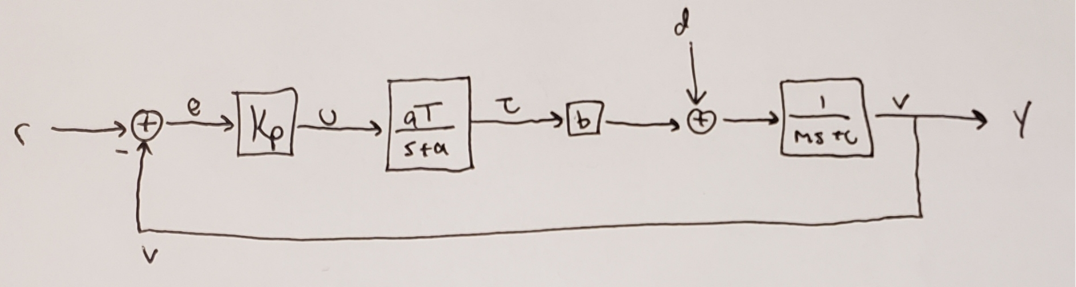

Figure 2: Block Diagram of Simple Cruise Control System

See MATLAB for the system simulation, computed via the transfer function, and the step response and bode plots of the system with $k_p = 0.01$ and $k_p = 0.1$, which is shown in Figure 3. As can be seen in the plot, the step response of the system with $k_p = 0.01$ has little oscillation, but has a rather large long term error, as it asymptotically reach 0.5 instead of the desired value of 1. Increasing k_p to 0.1 increases the oscillation and overshoot, but the long term error is decreased. The magnitude Bode plot now shows a peak.

(c) With a proportional and integral control law $u = k_p e + k_i x_c$ where $\frac{d}{dt}x_c = e = r - v$. This means that the transfer function for the control block is

$$\begin{aligned} u &= k_p e + k_i \int e \\ U(s) &= k_p E(s) + k_i \frac{1}{s} E(s) \\ H_U &= k_p + k_i \frac{1}{s} \end{aligned}$$

Again, the overall transfer function is computed in MATLAB. Figure 4 shows a block diagram of the system with this more complex control law, assuming $d = 0$.

(d) See the MATLAB code for the computation and plotting of the step and frequency response. Figure 5 shows these plots. The magnitude Bode plot now shows a strong peak, and the step response is highly oscillitory. Though the overshoot is now almost 100%, the long term error appears to go to zero. The addition of this integral control trades an increase in overshoot and oscillation for a reduction in long term error.

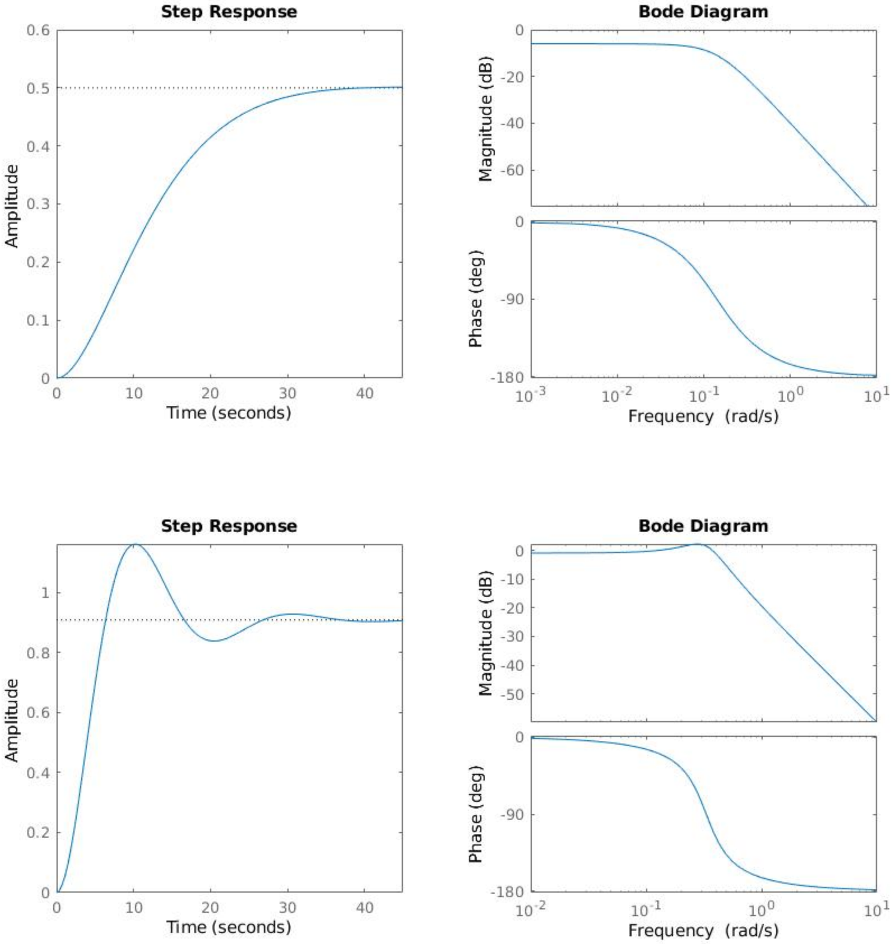

Figure 3: System characterization plots. On top is the step response and bode plot when $k_p = 0.01$, while the bottom shows the step response and bode plot for $k_p = 0.1$

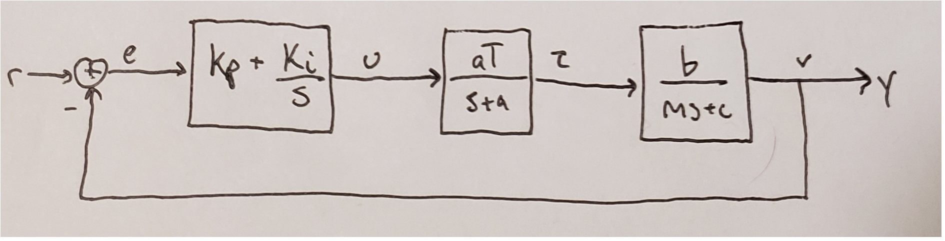

Figure 4: Block diagram of the system with the more complex control law

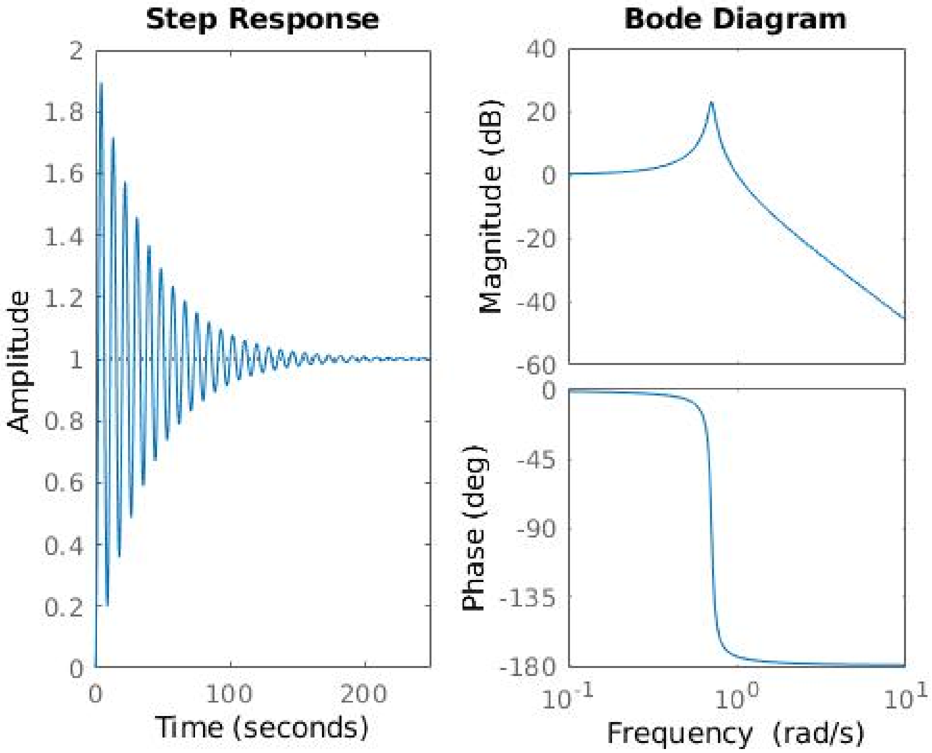

Figure 5: Step response and Bode plot for the systemw ith proportional and integral control law.