EEG307: Semester Review & Recap: Key Takeaways*

Lecture 36

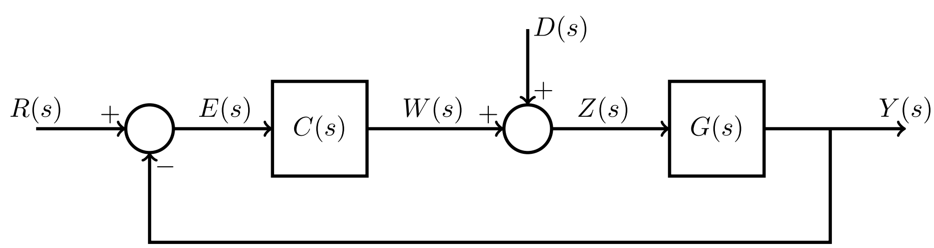
Elenya Grant, Kathryn Johnson, and Hisham Sager†

Fall 2022

Contents

1 Review of PID Control Structures

Recall that we have looked at variations of Proportional-Integral-Derivative (PID) controllers $C(s)$ within our standard negative unity feedback configuration with possible disturbance:



where the PID controller $C(s)$ is given by

$$C(s) = K_p + \frac{K_i}{s} + K_d s$$

In brief, the three terms of the PID controller accomplish the following:

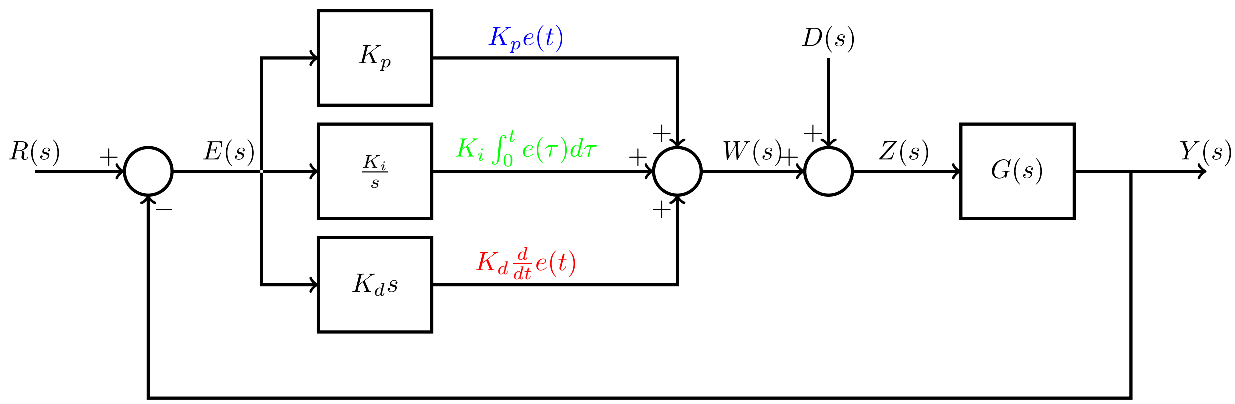
†Developed and edited by Tyrone Vincent and Kathryn Johnson, Colorado School of Mines, with contributions from Salman Mohagheghi, Chris Coulston, Kevin Moore, CSM and Matt Kupilik, University of Alaska, Anchorage

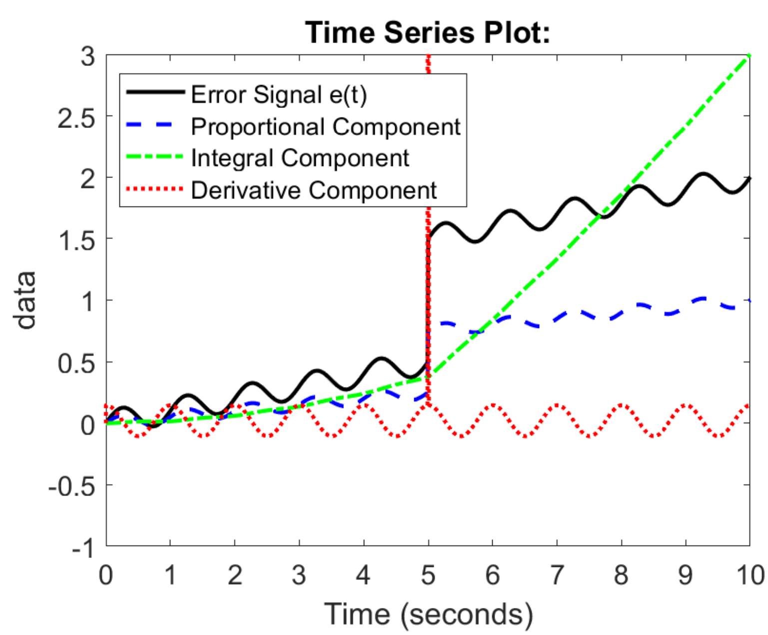
PID Term Impacts

- The **proportional term** K_p updates the control signal proportionally to the error
- The **integral term** K_i "accumulates" (integrates) the error over time so that if the error does not go to zero, the control signal keeps increasing in magnitude[1] to try to drive the error to zero
- The **derivative term** K_d, in a sense, tries to "predict" the future error by looking at the current slope of the error, which can help to smooth the response by making it less reactive

These concepts are illustrated in the figure below showing expanded PID control architecture and associated time series components for each of the three controller terms. This made-up error signal $e(t)$ contains sinusoidal, ramp, and step components and is plotted as the solid black line. The controller gains used to illustrate the components are $K_p = 0.5$, $K_i = 0.3$, and $K_d = 0.2$. *Note: there is no plant in the Simulink model used to produce this illustration and no feedback.*

Concept of PID control, expanded





Note in the figure that the error signal experiences a step increase at time $t = 5$ s, which leads to a large derivative component at that time. The integral component grows over time due to the nonzero error, most dramatically after the step in the error signal. The sinusoidal component of the error can be clearly seen in both the proportional and derivative components.

[1] could be a positive or a negative value

In this article, we will review how each of the four variations (P, PI, PD, and PID) of the PID control architecture impact

1. closed loop stability and stability margins,
2. the root locus plot,
3. the Bode plot, and
4. the time response

for an automobile system $G(s)$ with two variations: cruise control (i.e., control of the vehicle's velocity) and automatic parking (i.e., control of its position). Our goals are to understand how to match the controller structure to the time-domain response (both transient and steady-state) and stability margins (gain and phase margin).

Before we get started, let's consider another way of writing the controller transfer function that will be useful for our root locus and Bode analysis.

Knowledge Check

KC1 How would you write $C(s)$ in transfer function form to help us to understand the zeros and poles? In other words, combine the three terms over a common denominator and factor out a gain term.

$$C(s) = \frac{K_d s^2 + K_p s + K_i}{s} \tag{1}$$

$$= K_d \frac{s^2 + \frac{K_p}{K_d} s + \frac{K_i}{K_d}}{s} \tag{2}$$

$$= K_d \frac{(s + z_1)(s + z_2)}{s} \tag{3}$$

where z_1 and z_2 are the controller's zeros, which may be real or complex. Note by comparing (2) and (3) that $z_1 z_2 = \frac{K_i}{K_d}$ and $z_1 + z_2 = \frac{K_p}{K_d}$

2 The Automobile System

Recall the automobile cruise control example from Lecture 15, in which we had the goal of maintaining the desired speed even in the presence of disturbances. In that lecture, we found that the vehicle's velocity $v(t)$ was given by

$$V(s) = \frac{1}{ms + b} F(s) - \frac{1}{ms + b} \bar{D}(s)$$

where $f(t)$ is the force from the engine and we have re-defined the disturbance as $\bar{d}(t) \equiv -md(t)$, which is the opposing "disturbance" force due to the hill. (Doing so avoids having to use the additional m block that was incorporated into Lecture 15).

In terms of the feedback loop at the start of Section 1, for the velocity control problem we therefore have

$$\begin{aligned} G(s) = G_v(s) &= \frac{1}{ms + b} \\ &= \frac{Y(s)}{Z(s)} \\ Z(s) &= F(s) + \bar{D}(s) \\ W(s) &= F(s) \end{aligned}$$

Knowledge Check

KC2 The derived equations are for the velocity of the vehicle. What would the plant transfer function $G(s)$ be if we were interested in automatic parking, i.e., position control?

Since position is the integral of velocity, we have $X(s) = \frac{1}{s}V(s)$, and therefore

$$G_p(s) = \frac{1}{s(ms + b)}$$

For the purposes of illustrating class concepts, let's use some simple values for our car model and assume $m = 1000$ kg and $b = 100$ kg/s.

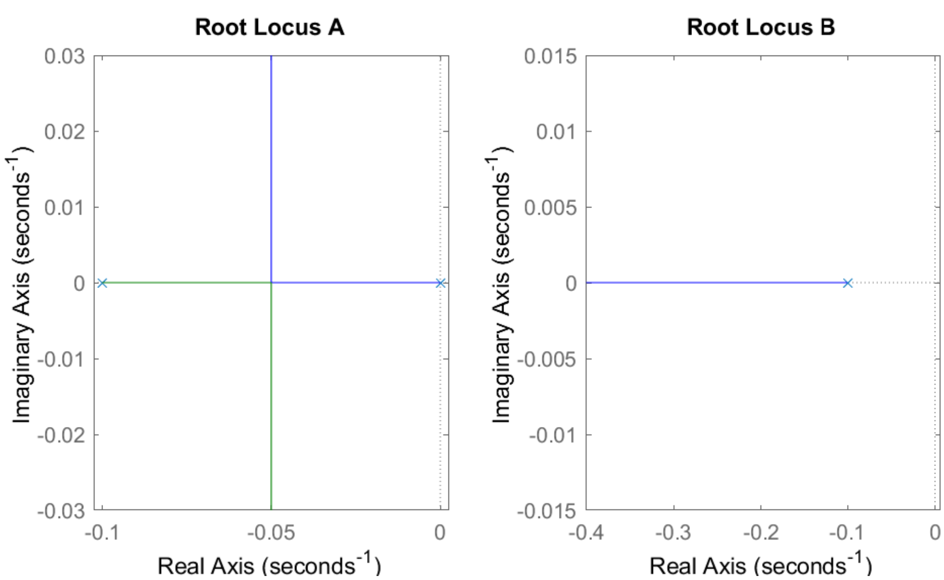
3 Root Locus Analysis

Recall that the root locus plot allows us to visualize the set of possible closed-loop poles, i.e., poles of the transfer function

$$T(s) = \frac{Y(s)}{R(s)}$$

for a system with a given $C(s)G(s)$ as a gain varies. If we first consider only proportional control, then $C(s) = K_p$, and the root locus plots for $K_pG_v(s)$ and $K_pG_p(s)$ are shown below.

Root Locus for Vehicle System with Proportional Control



Knowledge Check

KC3 Which root locus plot corresponds to the position control system $G_p(s)$ and which corresponds to the velocity control system $G_v(s)$?

Note that the proportional controller has no zeros or poles, so the only poles come from the plant $G(s)$. Since the position control system $G_p(s)$ has two poles, it corresponds to the left figure, "Root Locus A", which has a pole at zero and one at $-\frac{b}{m} = -0.1$.

Knowledge Check

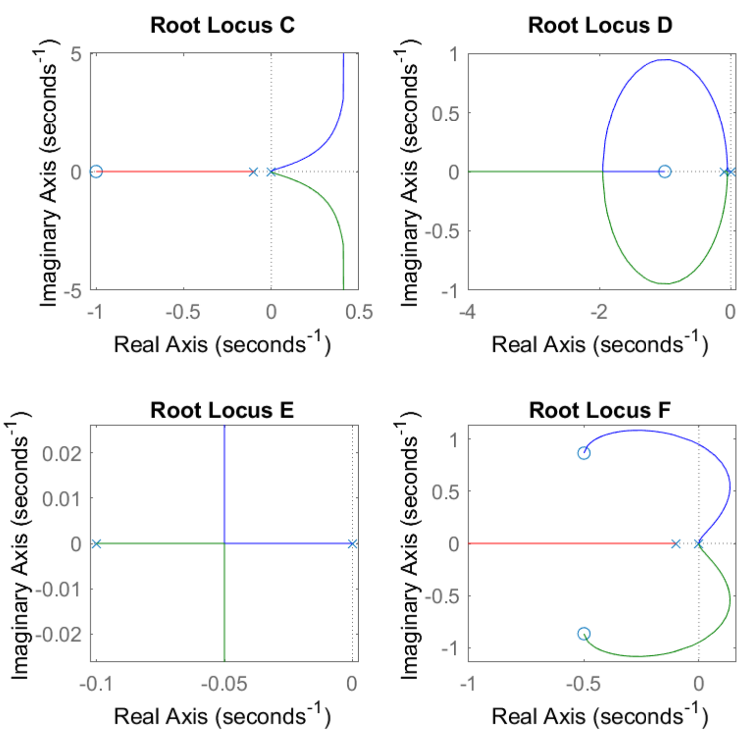
KC4 Given the two root locus plots, which is more likely to achieve a small %OS specification in response to a step reference input across a wide range of controller values K_p?
KC5 For both plots, are there any values of the proportional gain $K_p > 0$ that would result in an unstable closed-loop system?

For "Root Locus A" (the position control case), as K_p gets large, the closed-loop poles will depart from the real axis, so a nonzero $\%OS$ is possible. For the velocity control case shown in "Root Locus B," the closed-loop poles are always on the negative real axis, so the closed-loop $\%OS = 0$.

In both cases, the closed-loop poles are in the left half-plane (LHP) for all values of $K_p > 0$, so the closed-loop system is stable.

The figure below shows root locus plots for the position control system $G_p(s)$ with four variations of the PID controller: P, PI, PD, and PID. Each starts with equation (2) and $K_p = K_i = K_d = 1$, but then the relevant values are multiplied by 0 to get the appropriate PID variant. In other words, for the PD controller, we'd use $0 * K_i$ but keep $K_p = K_d = 1$.

Root Locus for Vehicle Position System with Four Controllers



Knowledge Check

KC6 What are the controller pole(s) and/or zero(s) in each of the PD, PI, and PID cases?
KC7 By considering these pole and/or zero values, which of the four root locus plots corresponds to which controller?
KC8 By considering the loci locations in the LHP and RHP, are there any values of the gain $K_d > 0$ (see (3)) that would result in an unstable closed-loop system for each of the four root locus plots?

Poles/Zeros: For the PD controller, there are no poles and one zero at $-\frac{K_p}{K_d}$. For the PI case, there is a pole at 0 and a zero at $-\frac{K_i}{K_p}$. For the PID controller, there is a pole at 0 and zeros at the roots of s^2+s+1, which are $-0.5 \pm j0.866$.

Matching Plots: By comparing the four plots in this figure to the earlier root locus figure, it's clear that "Root Locus E" is the same as "Root Locus A," so this is the case with the proportional controller. Next, note that "Root Locus F" has two complex zeros, so from the knowledge check above it must correspond to the PID controller, since both the PI and PD controller have a single real zero.

It's harder to distinguish between the PI and PD controllers because both have a single real zero, with the distinguishing feature being the PI controller's pole at 0. However, the plant $G_p(s)$ also has a pole at 0, and Matlab doesn't draw multiple poles at the same location. In other words, we are looking at these two similar situations:

$$C_{PI}(s)G_p(s) = K_p \frac{s+\frac{K_i}{K_p}}{s^2(ms+b)} \tag{4}$$

$$= \frac{s+1}{s^2(1000s+100)} \tag{5}$$

$$C_{PD}(s)G_p(s) = K_d \frac{s+\frac{K_p}{K_d}}{s(ms+b)} \tag{6}$$

$$= \frac{s+1}{s(1000s+100)} \tag{7}$$

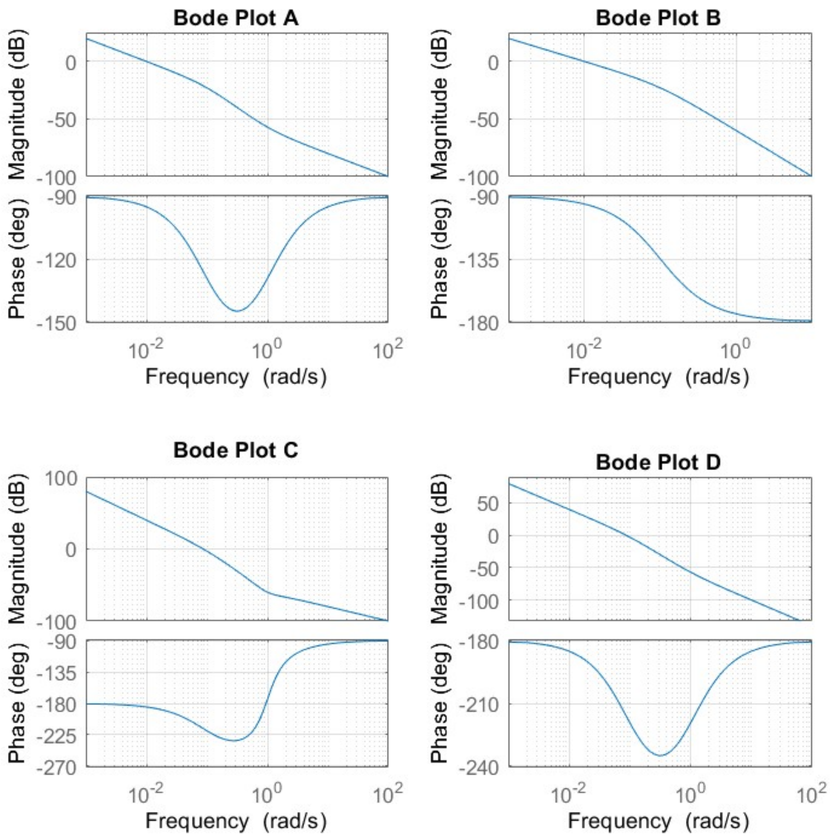
Thus, we must use our root locus knowledge to distinguish between these cases. In particular, we know that loci on the real axis lie to the left of an *odd* number of real poles and zeros, so in the case with just one pole at the origin in (7). In other words, the root locus plot that matches $C_{PD}(s)G_p(s)$ must have a locus on the real axis to the left of the origin, which corresponds to "Root Locus D" (though it's a little hard to see). By process of elimination, then, "Root Locus C" is for the PI control case.

Stability: Finally, we consider stability. Both "Root Locus D" and "Root Locus E" have all loci in the LHP and are therefore closed-loop stable for any gain value we select. "Root Locus F" (the PID controller) has two loci that enter the RHP for small values of K_d, but then trend back into the LHP as K_d gets larger.[2] Thus, the PID controller is appropriate to consider as long as care is taken in the tuning of K_d. Finally, the PI controller shown in "Root Locus C" has two loci in the RHP for all possible choices of K_p, which means that PI control is not a suitable choice for this specific system.

4 Bode Analysis

Let's now consider a similar set of questions from a Bode perspective. Let's start by looking at four Bode plots for one of the vehicle systems (either $G_p(s)$ or $G_v(s)$) and all four variations of the PID controller that we reviewed above, namely P, PD, PI, and PID.

Bode Plots for Vehicle System with Four Controllers



Knowledge Check

KC9 Given that all four plots are for the same plant system, were they made with the position control system $G_p(s)$ or the velocity control system $G_v(s)$? *Hint: consider the low-frequency slope of the magnitude plot or the low-frequency phase angle.*

KC10 Which Bode plot corresponds to the P, PD, PI, and PID controller?

KC11 Given your knowledge of system Type and steady-state error (see Lecture 16), which Bode plots indicate that we can achieve zero steady-state error in response to a step reference input?

Which plant?: Note that none of the Bode plots have a low-frequency magnitude slope of 0 dB/dec, which is the slope we would expect for a Type 0 system such as $G_v(s)$ combined with a proportional controller (neither of which have a pure integrator, i.e., a pole at 0). Therefore, these plots must have been generated with the position control plant system $G_p(s)$.

Matching plots: There are a couple of ways to approach this matching problem. Since it's a bit hard to read the magnitude plot slopes exactly, it might be easier to start with the phase plots. "Bode Plot A" and "Bode Plot B" both start at $-90°$, which means they have one integrator each (from the plant $G_p(s)$) and are therefore associated with the P and PD controllers, whereas "Bode Plot C" and "Bode Plot D" start at $-180°$ phase and therefore must include the integral term with another pole at 0 (i.e., the PI or PID controller).
To distinguish among "Bode Plot A" and "Bode Plot B," note that a LHP zero from the PD controller would result in a net phase change of $+90°$ compared to the P controller. Thus, "Bode Plot A" corresponds to the PD controller and "Bode Plot B" corresponds to the P controller case.
We can use the same method to distinguish between "Bode Plot C" and "Bode Plot D," namely, that the PI controller has a single LHP zero and the PID controller has two. Thus, the PID controller results in a net phase change of $+90°$ compared to the PI controller, which means that "Bode Plot C" has the PID controller and "Bode Plot D" has the PI controller.

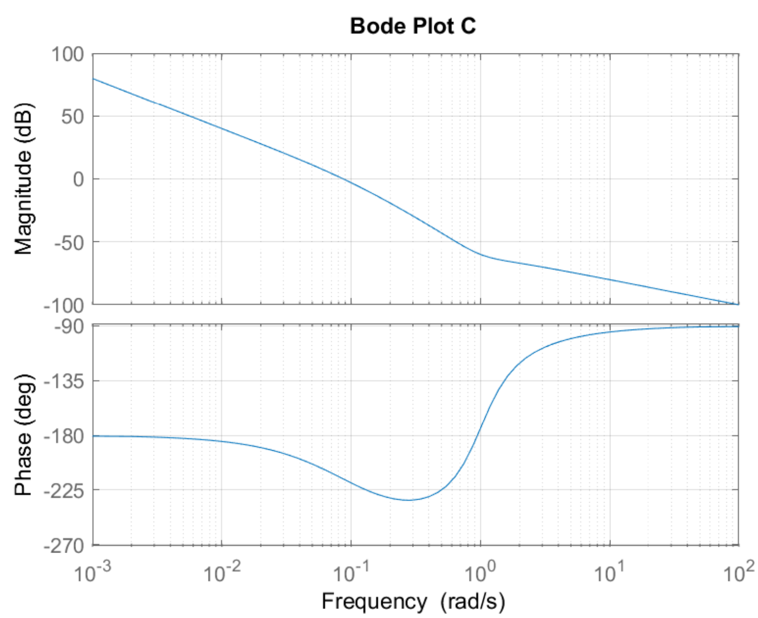
Steady-state error: If you looked at the low-frequency magnitude plot slopes, noticed that all are at least -20 dB/dec (indicating a Type 1 system) or higher, and therefore concluded that all four cases could achieve zero steady-state error to a step reference, you'd have missed one crucial step: *checking for stability*. Recall from Section 3 that the PI controller case is always unstable in closed-loop when combined with this plant $G_p(s)$, so this case will never result in zero steady-state error in closed-loop.
Checking closed-loop stability for the other three cases, and keeping in mind that none of the open-loop controller variations or plant $G_p(s)$ have poles in the open RHP (i.e., real part greater than zero), we can estimate the following:

Bode Plot	Gain Margin	Phase Margin	Stable?
A	∞	$\approx 85°$	GM>0, PM>0 $\Rightarrow$ Yes
B	∞	$\approx 85°$	GM>0, PM>0 $\Rightarrow$ Yes
C	undefined	$\approx -35°$	PM<0 $\Rightarrow$ No

Thus, only Bode plots A and B will result in zero steady-state error to a step reference (since these are the only two stable configurations).

Let's take a closer look at the PID control case wherein stability could be altered by appropriate re-tuning of the gain K_d (see (3)). Consider the zoomed-in plot for "Bode Plot C."

Bode Plot for Vehicle Position System with PID Controller

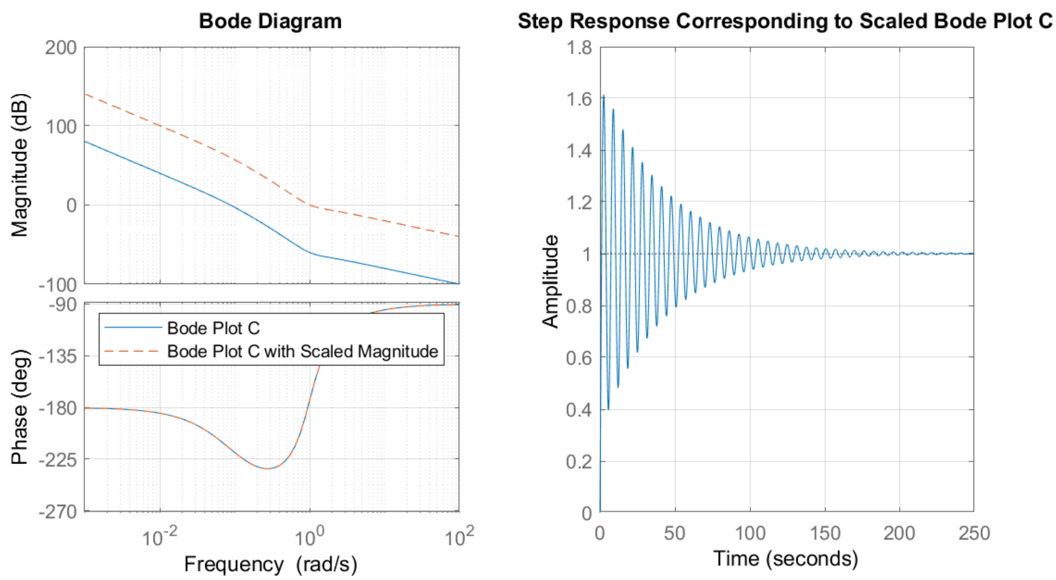


Knowledge Check

KC12 By how much would we need to increase the controller gain to create a stable closed-loop system (and thus get zero steady-state error to a step reference, since the system is already Type 2)?

Gain for stable closed-loop: From the zoomed-in plot, it appears that the crossover frequency ω_{co} at which the phase margin is measured must be above $10^0 = 1$ rad/s for the phase margin to be positive. To achieve $\omega_{co} = 1$ rad/s, we must increase the magnitude Bode plot by 60 dB, or a gain $K = 10^{\frac{60}{20}} = 1000$. The figure below shows the original and scaled Bode plots on the left and the time series response on the right. Although the response is very oscillatory due to the fact that the phase margin is very small (so the closed-loop damping ratio ζ_T is also very small[3]), it is at least consistent with a closed-loop stable response and approaches zero steady-state error.

Bode Plot and Step Response for Scaled PID Controller

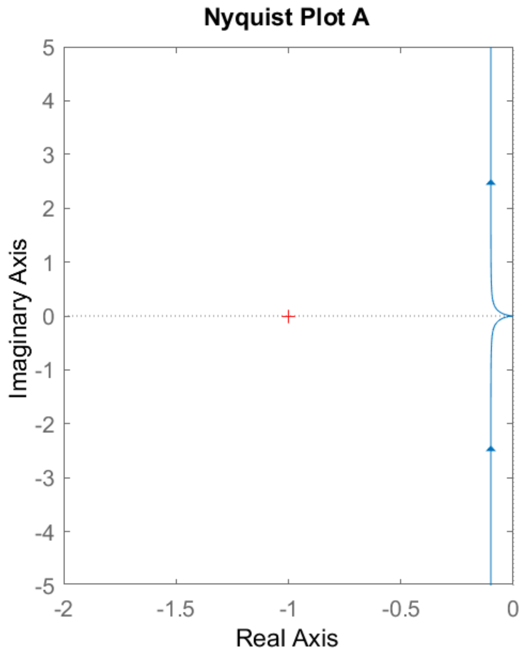


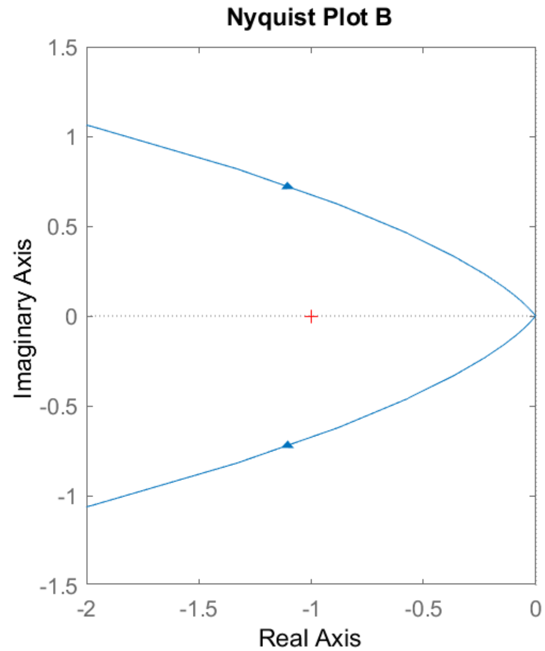
5 Nyquist Analysis

Next, let's take a look at the vehicle position control system from a Nyquist perspective. Again, recall that neither the plant $G_p(s)$ nor any of the controller variations have poles in the open RHP, so the Nyquist term $P = 0$ for all cases.

Two Nyquist plots corresponding to the P and PID controllers acting on $G_p(s)$ are shown in the figure below.

Nyquist Plot for Vehicle Position System with PID Controller





Knowledge Check

KC13 What do you need to do to complete each Nyquist plot? *Hint: recall that Matlab can't draw portions of the plot associated with poles on the imaginary axis*

KC14 Which Nyquist plot corresponds to the P controller and which corresponds to the PID controller (both with plant $G_p(s)$)?

KC15 After completing the plots, use Nyquist criteria to determine the number of unstable closed-loop poles Z in each case.

KC16 Verify your answer to the number of unstable closed-loop poles by calculating the closed-loop transfer function $T(s)$ and its poles.

Completing the plots: We know from Lecture 29 that, for each pole on the imaginary axis, we need to add a clockwise half-circle at radius infinity to the Matlab-generated Nyquist plot. By reviewing the two plots, it looks like "Nyquist Plot A" could be completed with one such half circle (indicating a single pole on the imaginary axis) and "Nyquist Plot B" could be completed with two such half circles (indicating two CW half circles).

Matching plots: After determining that "Nyquist Plot A" will have one CW half circle, we know it must correspond to the P controller since $C_P(s)G_p(s)$ has just one pole on the imaginary axis (at the origin). Thus, "Nyquist Plot B" corresponds to the PID controller $C_{PID}(s)$.

Unstable closed-loop poles: "Nyquist Plot A" has zero encirclements of $-1 + j0$, so $N = 0$. Thus, $Z = N + P = 0$ and the closed-loop transfer function $T(s)$ has zero poles in the RHP. By noting the arrow directionality from Matlab in "Nyquist Plot B," we see that it has two encirclements of $-1+j0$, meaning that $N = 2$, and therefore $Z = N+P = 2$ and $T(s)$ has two poles in the RHP.

Checking closed-loop poles: In both cases, the closed-loop poles can be found from the roots of the denominator of

$$T(s) = \frac{C(s)G(s)}{1 + C(s)G(s)}.$$

In the PD case, we have

$$T(s) = \frac{1}{1000s^2 + 100s + 1}$$

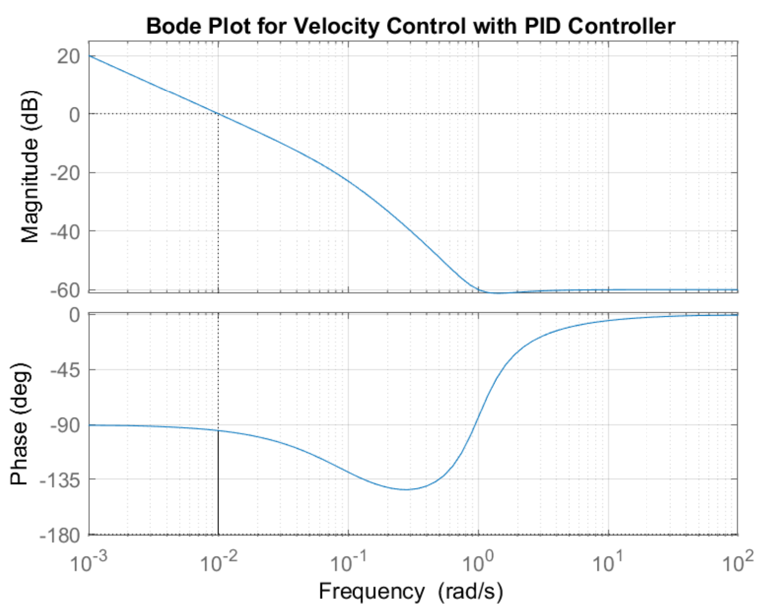
which has closed-loop poles -0.09 and -0.01, both of which are in the LHP, confirming our "Nyquist Plot A" stability analysis.

$$T(s) = \frac{s^2 + s + 1}{1000s^3 + 101s^2 + s + 1}$$

which has closed-loop poles $-0.14, 0.02 \pm j0.08$, confirming our "Nyquist Plot B" analysis that there are two (unstable) poles of $T(s)$.

6 Impact of Time Delay

Now that we've refreshed our understanding of the various tools for the vehicle position control system, let's take a look at velocity control for $G_v(s)$ in the case of a time delay, for example in the case of a low-performance car with some poor connections in the feedback loop that cause information to take too long to travel or a slow actuator (throttle). The figure below shows the case where PID control is used. It was produced using the Matlab command `margin`, which automatically draws the vertical black line indicating PM (as well as GM in cases where $GM \neq \infty$).



Knowledge Check

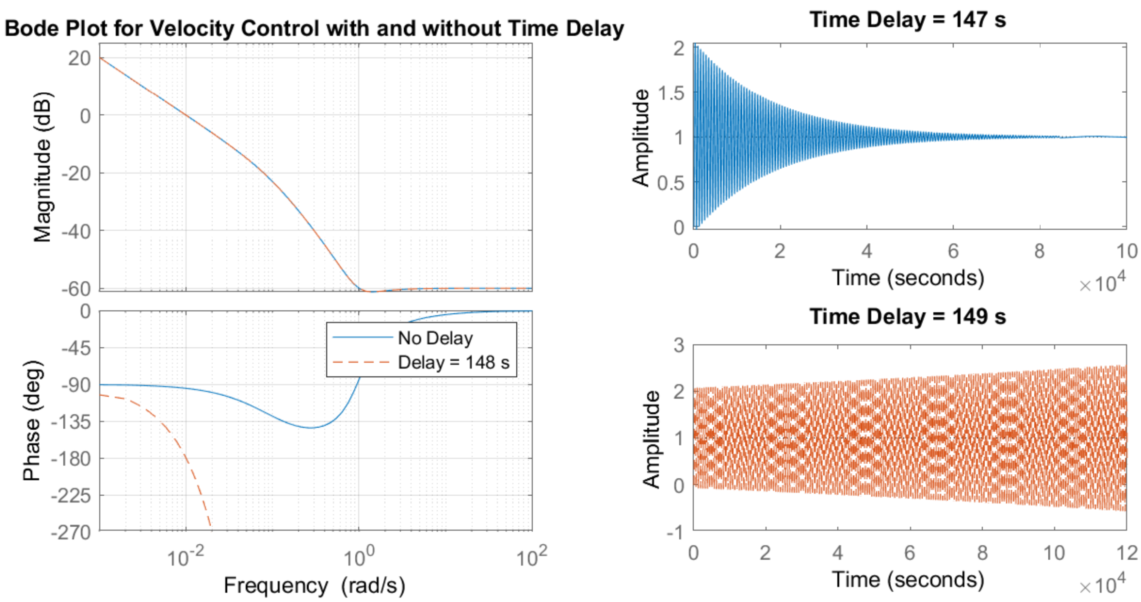
KC17 What is the maximum time delay the system can tolerate before going unstable? Verify your answer by plotting the time response to a step reference for time delay values just under and just over the calculated amount.

KC18 In the presence of this calculated time delay, how would you need to change your controller's gain to achieve $\%OS = 16.5\%$?

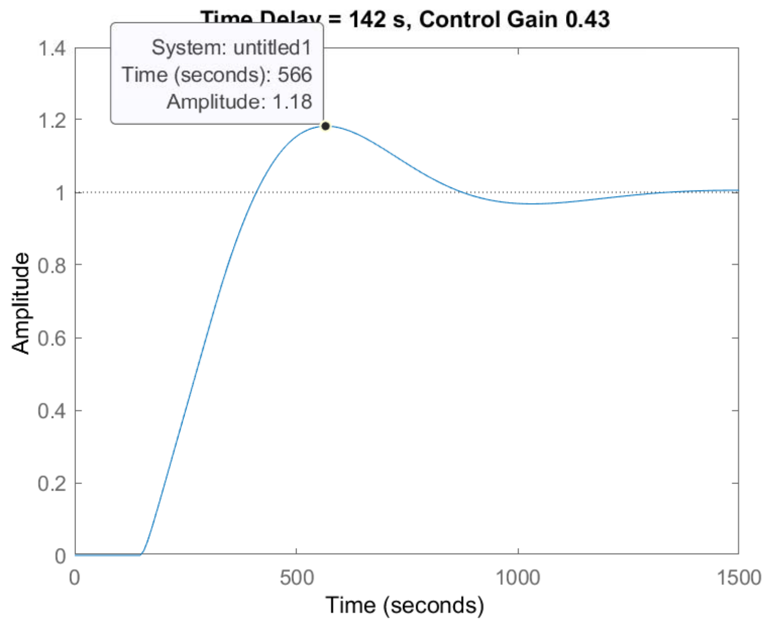
Max time delay: Using the "Data Tips" tool in Matlab, we see that the phase margin $\phi_{PMG} \approx 85°$ at a frequency of $\omega_{co} = 0.01$ rad/s. Thus,

$$T = \frac{\phi_{PMG}\left(\frac{2\pi\text{rad}}{360°}\right)}{\omega_{co}} \approx 148\text{s}.$$

The Bode plot with the calculated allowable time delay $T = 148$ s and time responses with delays just under (147 s) and just over (149 s) that value are shown below. Since the 147-s delay appears to converge and the 149-s delay appears to diverge, these plots are consistent with our calculation.



$\%OS$: From the example in Lecture 32, a closed-loop $\%OS = 16.5\%$ corresponds to a closed-loop damping ratio of $\zeta_T = 0.5$. Thus, we need a phase margin $\phi_{PMG} = 50°$. In the delayed Bode plot above (red dashed line), the phase is $-130°$ at a frequency of ≈ 0.004 rad/s. Thus, we need to *reduce* the magnitude so that the new $\omega_{co} = 0.004$ rad/s, which means we need to reduce it by -7.2 dB or a scalar gain of $10^{\frac{-7.2}{20}} \approx 0.43$. The resulting step response is shown below. Note that the $\%OS$ is slightly high, but within the margin of error for the approximations used (both in reading the Matlab plot and in connecting $\%OS$ to ϕ_{PM}) as well as for the fact that there are zeros in the closed-loop system that are not accounted for.



7 Disturbance Rejection

Finally, let's check the ability of our velocity control system with PID controller to reject disturbances (e.g., due to a hill). Recall that the output signal $Y(s)$ for a feedback system with multiple inputs is a function of both inputs, i.e.,

$$Y(s) = T_1(s)R(s) + T_2(s)D(s)$$

and that we can find the two closed-loop transfer functions $T_1(s)$ and $T_2(s)$ by setting the other input to zero since we are working with linear systems for which superposition applies.

Similarly, for disturbance rejection we often want to calculate a different closed-loop transfer function $T_3(s) = \frac{E(s)}{D(s)}$ so that we can design the controller to ensure the steady-state error *caused by* the disturbance is $e_{ss} = 0$. To find $T_3(s)$ we would also set the other input signal $R(s) = 0$.

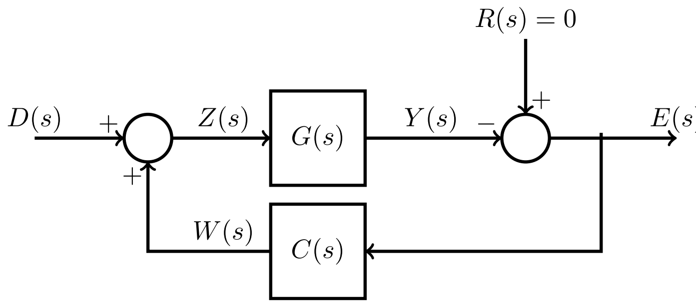
Knowledge Check

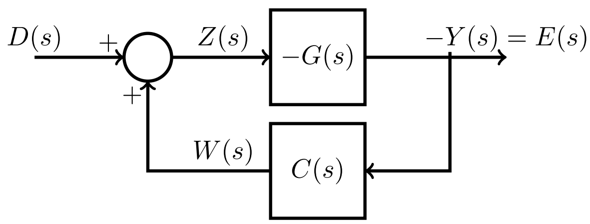
KC19 Sketch the feedback loop from disturbance $D(s)$ to output $E(s)$ when $R(s) = 0$ in the familiar form, i.e., with $D(s)$ shown as the input signal at the left side and $E(s)$ shown as the output signal at the right side. Do not change any of the signal-system relationships, just where they are positioned on the page. Label all of the signals and systems that are shown in Section 1.

KC20 Use your reconfigured feedback loop to find the closed-loop transfer function $T_3(s) = \frac{E(s)}{D(s)}$.

KC21 What restrictions, if any, must be placed on a PID controller $C(s)$ to ensure that a step disturbance would be fully rejected ($e_{ss} = 0$)?

Reconfigured feedback loop: Two versions of the reconfigured loop are shown below, the first with all signals shown and the second with $R(s) = 0$ removed. Notice that in that second case the $-$ sign that was associated with the summing junction at the right has been absorbed into the plant block.





Closed-loop transfer function: By recalling the rule that a feedback loop can be simplified as the forward gain divided by one minus the loop gain, and noting that we're using the velocity variation of the plant system $G_v(s)$ in this section, we have

$$\begin{aligned} T_3(s) &= \frac{E(s)}{D(s)} \\ &= \frac{-G_v(s)}{1 - (-G_v(s)C(s)} \\ &= -\frac{G_v(s)}{1 + G_v(s)C(s)} \end{aligned}$$

PID control stability and disturbance rejection: Using the PID controller form from (1) and the velocity version of the plant $G_v(s)$, we can find that

$$T_3(s) = \frac{-s}{(m + K_d)\, s^2 + (b + K_p)\, s + K_i}$$

First, we need to ensure that the closed-loop system is stable, which for a 2nd-order system requires that all coefficients on s in the denominator be greater than zero (remember that this trick doesn't work for higher-order systems). Thus, to ensure zero steady-state error for a step disturbance, we *first* need to ensure stability, namely, we need $m + K_d > 0$, $b + K_p > 0$, and $K_i > 0$.

Then, we can apply the Final Value Theorem (Lecture 15). The FVT tells us that, assuming the system is stable, the final value of a signal can be found by

$$e_{ss} = \lim_{t\to\infty} e(t) = \lim_{s\to 0} sE(s).$$

In our case, this means that

$$\begin{aligned}
e_{ss} &= \lim_{s\to 0} s \frac{-s}{(m + K_d)\, s^2 + (b + K_p)\, s + K_i} D(s) \\
&= \lim_{s\to 0} s \frac{-s}{(m + K_d)\, s^2 + (b + K_p)\, s + K_i} \left(\frac{A}{s}\right) \\
&= \lim_{s\to 0} \frac{-As}{(m + K_d)\, s^2 + (b + K_p)\, s + K_i} \\
&= 0
\end{aligned}$$

(where A is the amplitude of the step disturbance) since $K_i > 0$ from the stability restriction. Thus, as long as the closed-loop system is stable, step disturbances will be successfully rejected for the velocity control plant $G_v(s)$ with PID controller.

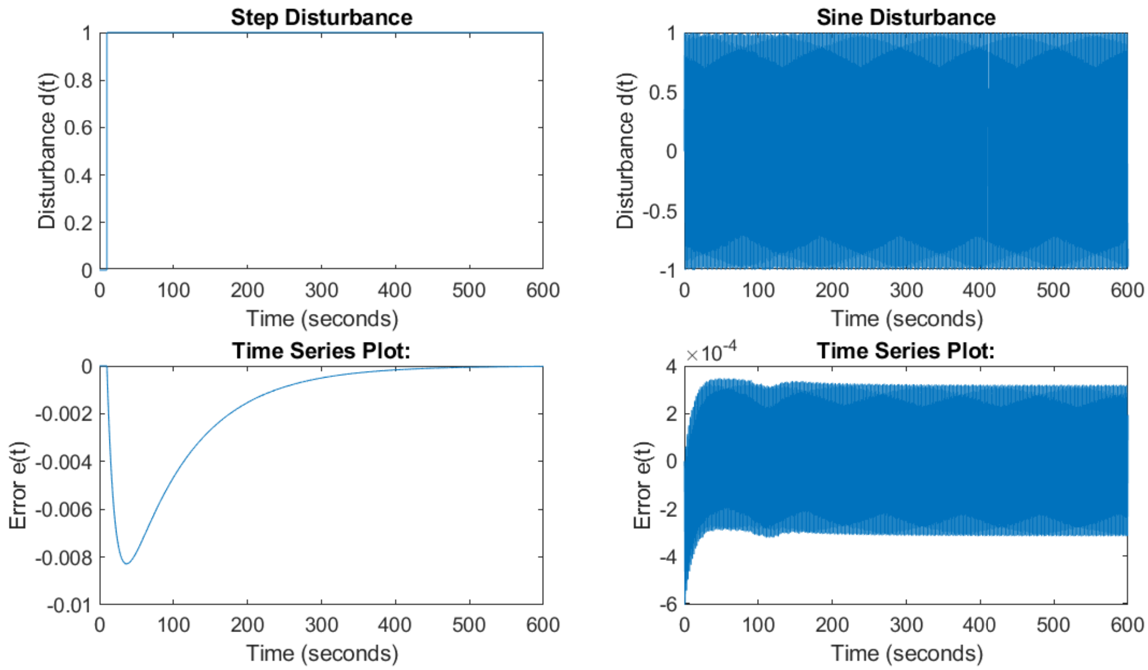
Regarding the disturbance rejection analysis, we note that a similar analysis can be performed for other types of disturbance signals, where $D(s)$ would be replaced by the Laplace transform of the desired disturbance signal. A key point is to remember *whether a final value exists*, which depends on whether the output signal of interest (e.g., $e(t)$ or $y(t)$) converges to a final value or whether it diverges (e.g., for an unstable system) or oscillates (e.g., for a constant-amplitude sinusoid). See (Lecture 15) for details.

If a final value does not exist, it's still possible that the error could be bounded and remain within an acceptable value. One way to check this is to find $E(s)$, then use inverse Laplace transform techniques to find $e(t)$. Tools like Simulink can also be used to perform this analysis.

Knowledge Check

KC22 Use Simulink to find the steady-state error $\lim_{t\to\infty} e(t)$ for the velocity control plant system $G_v(s)$ in response to (a) a unit step disturbance $d(t) = u(t)$ and (b) a sinusoidal disturbance $d(t) = \sin(\pi t)u(t)$.

Error from Simulink: Simulink-generated results for both cases are shown in the figure below.



These plots use the same $K_p = K_d = K_i = 1$ as earlier analysis in this lecture, and clearly the controller could use a tuning for better performance as the response is very slow. However, the left plot does verify that our steady-state error approaches zero in response to a step disturbance. The right plot indicates that the error in response to a sinusoidal disturbance is also sinusoidal (as would be expected), though it is difficult to read on the time scale used. The amplitude of $e(t)$ is about three orders of magnitude smaller than the amplitude of $d(t)$, which is as would be expected given $G_v(s)$. If desired, we could determine the precise reduction in amplitude using Bode concepts in Lectures 24-27.