# University of Utah
# Electrical & Computer Engineering Department

ECE 3510 Lab 5a

# Velocity Control and Steady - State Error

M. Bodson, A. Stolp, 2/12/08
rev, 2/12/09, 2/15/17 Bhavana Mukunda, 2/20/19 Abid Hossain

**Note:** Bring the lab-2 handout to use as a dSPACE tutorial. Also bring an eraser like you would use to erase pencil lines. You will use it as a brake on the motor shaft.

## Objectives

- Create an angular-velocity control system.
- Observe the steady-state error in the measured velocity step responses.
- Use feed-forward compensation to reduce the steady-state error.
- Use proportional-integral control to reduce the steady-state error.

## Equipment and materials from stockroom:

- DC Permanent-magnet Motor (If you can, get the same motor you used last time.)
- Dual Power Amp
- dSPACE cable

## Experiment 1, Proportional Velocity Control

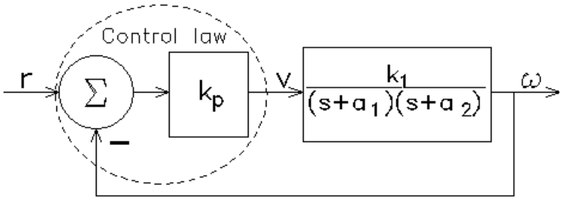
This lab will use a simplification of the model of the DC motor that you found in lab 4. In this lab, the motor will be used in an closed-loop angular-velocity control system. In order to analyze the loop, we'll need a simple transfer function for the motor, that means, excluding the coulomb friction factors. Look back at lab 4 now and find your number for $B_{m2}$, the $B_m$ that sort-of models both types of friction in one. Recalculate $a_1$ and $a_2$ with this $B_m$ and your best numbers for all the other parameters. They should be about 80 $sec^{-1}$ and 650 $sec^{-1}$, respectively. Find your $k_1$. If it includes $v_a$, the input voltage, recalculate without $v_a$. It should be about 500,000 $rad/Vsec^3$. Record these numbers for future calculations and write your motor transfer-function as:

$$P(s) = \frac{\omega(s)}{V(s)} = \frac{k_1}{(s+a_1)(s+a_2)}$$

In the simplest type of control system the controller is just an amplifier with simple proportional gain. This is sometimes called a "proportional control law". And can be expressed like this:

$$v = k_p(r - \omega)$$

Notice that this represents more than just one block in the block diagram.



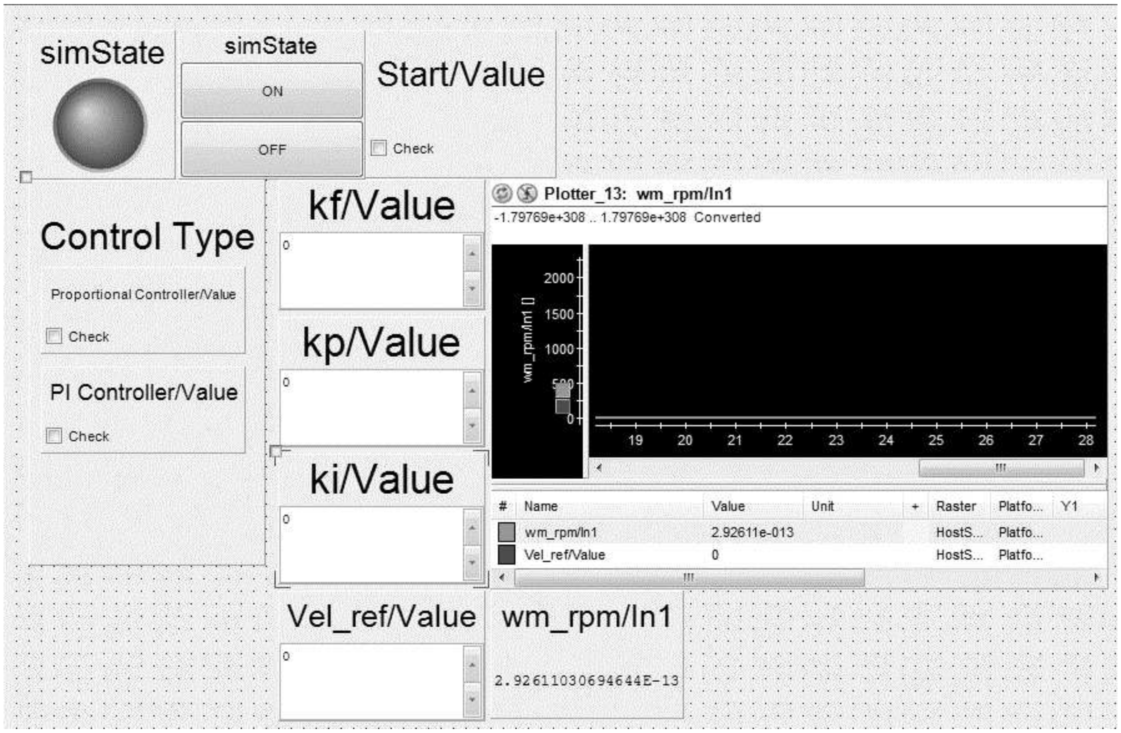
The output of this loop is $\omega$, the angular velocity in rad/sec. The input is labeled r, for "reference" or "reference input". It is also in rad/sec (even though the control panel happens to be in rpm). This is a little different from the terminology you may be used to in class. In your notebook, use the general loop transfer function to derive the closed-loop transfer function:

$$H_{CL}(s) = \frac{\omega(s)}{r(s)} = \frac{k_P k_1}{(s+a_1)(s+a_2)+k_P k_1} = \frac{k_P k_1}{s^2+(a_1+a_2)s+(a_1 a_2+k_P k_1)}$$

Now find the value of $k_p$ for critical damping.

Next, find the value of $k_p$ that would result in two complex poles where the magnitudes of the real and imaginary parts are equal. This is not as hard as it sounds, it's just like what you did above, except this time the part of the quadratic equation under the radical sign is not 0. Instead, it should be $-(a_1+a_2)^2$. Like critical damping (which will result in the fastest response with absolutely no ringing), this condition is considered a special placement for the poles. When the real and imaginary parts are equal in magnitude, the damping factor ( $\zeta$ ) = 0.707, the overshoot is about 4% (usually acceptable) and the response is a bit faster.

All right, let's build this thing! Hook up the dSPACE system, the amp, and the motor as you have several times before.
Navigate to the Labs page found under www.ece.utah.edu/~ece3510 and download: PI_Control_DC_Motor_S19.zip (or whatever is shown there for lab 5a). Extract the zip file. Refer to lab 2 handout section: **dSPACE Control Desk Setup** to open dSPACE ControlDesk, create a new Project+Experiment, and import the .sdf file provided. The layout is shown below.



kf/Value: 0

kp/Value: 0

ki/Value: 0

Vel_ref/Value: 0

wm_rpm/In1: 2.92611030694644E-13

| # | Name | Value | Unit | + | Raster | Platfo... | Y1 |
|---|---|---|---|---|---|---|---|
| ▨ | wm_rpm/In1 | 2.92611e-013 | | | HostS... | Platfo... | |
| ■ | Vel_ref/Value | 0 | | | HostS... | Platfo... | |

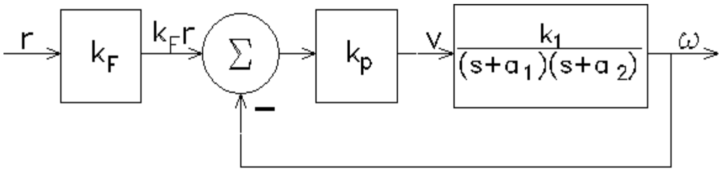
Engage the layout online and turn ON the experiment. Check the "Start" box. When "Control type" is set to "Proportional" only the "Speed (Rpm)", "Proportional gain" and "Feed-forward gain" parameters are used.

Set the "Feed-forward gain" to 1. Set the "Proportional gain" to the $k_p$ for critical damping. Clear the motor area, hit "Run" and then use the little up and down arrows next to the "Speed (Rpm)" to change the speed from 0 to 1000 rpm, 2000 rpm and then back down to 1000 rpm and 0. Two steps up and two steps down. (You may want to change the step size (to 1000) in the instrument properties under "Increments".) Once the Vel_ref is reduced to 0, go through the following procedure to turn off the experiment so you can set up the recording. 1. Uncheck the Proportional controller in the "Control Type" box to turn off all output signals. 2. Uncheck the "Start" button on the layout. 3. Click "Off" 4. Go offline. 5. Turn off the dual amp if you're not continuing right away.

Refer to lab 2 handout section: **Setting up data recording** for further setup. Use Trigger Rule 1.txt for the start condition. The Stop condition can be set to ~ 8 second TimeLimit or Trigger Rule 2 or None and you can stop manually.

Go back Online, hit "Start" and "Proportional Controller". Check your gains, hit "Start Triggered Recording" and then repeat the rpm steps to record a data set to make a plot for your notebook. The plot should show both the reference (input) and the output velocities. The fuzz on the plot is caused by the computer sampling of the velocity. Any overshoot or ringing is being caused by something we aren't modeling.

Notice that the measured speed doesn't match the reference speed very well. This is called *steady-state error* for (hopefully) obvious reasons. To see why there's so much error, calculate the DC gain of the closed-loop transfer function ($H_{CL}(0)$). Pretty cruddy, huh? No wonder the output is only a little over half the reference. Not good... How can this be fixed? Increasing the proportional gain will help, but that would make the system underdamped. What if we just increase the reference input a little to compensate for the fact that the output doesn't reach the reference? How about increasing it by $1/H_{CL}(0)$ to exactly compensate for the cruddy DC gain? This type of gain is called *feedforward* gain. It is covered on page 78 of your Bodson text, and we have an input for it on our screen. Set that input to the necessary gain now and take another data set for another plot. Does it work?



Unfortunately feedforward compensation (of which feedforward gain is a simple example) requires a good knowledge of the system and doesn't adjust when the system changes or suffers disturbances. Run the system again at a constant rpm. Gently touch your eraser to the coupler in such a way that the coupler teeth won't grab it. This introduces a braking effect and is thus a *disturbance* to the system. Try to keep the eraser pressure constant. Does the system maintain or regain the reference input velocity? Comment in your notebook, but don't bother with a plot.

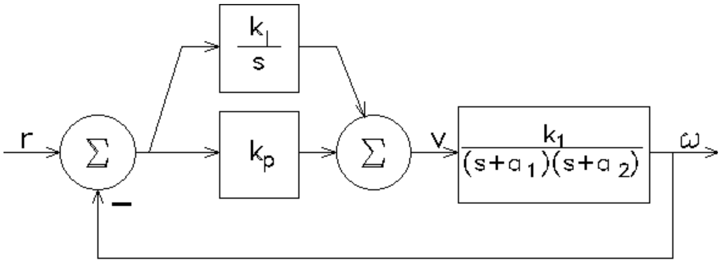
Set the $k_p$ to the value you calculated would give two complex poles and $\zeta = 0.707$. Set $k_F = 1$. Repeat the steps above, except for the eraser part.

You'll notice that the overshoot and ringing are already larger than expected, but nonetheless, try to turn up the gain even higher to reduce the steady-state error without feedforward gain (set to 1). How far can you go before you get significant oscillations? Remember, always be sure to uncheck the controller after Vel_ref is down to 0 to stop the motor from ringing. You will have to check it again when you need to run the proportional controller again.

Gosh, this gain ($k_p$) is a pretty important parameter. On the plus side, increasing the gain increases the speed of response and decreases the steady-state error. On the negative side, increasing the gain decreases the stability of the system. Wouldn't it be nice if we had a good tool to predict how the system response is affected by the gain? The root-locus method is exactly the tool we're looking for and you'll see it soon in class.

## Experiment 2, Proportional - Integral Velocity Control

So far we've seen two ways to decrease the steady-state error. Add gain outside the loop or add gain inside the loop. Both methods had limitations. Now we're going to look at a third way.



The steady-state error shows up as a signal at one point in the loop, just after the summation (subtraction) circle. In fact, that signal is the error at all times, not just in the steady-state condition. If we integrate that error signal, even a small error would soon integrate to a large value. If the integrated error were part of the signal fed to the motor, the error would have to eventually go to zero. This method is called integral-proportional (or PI) control. We put the two together because integral control alone is too slow.

Find the new transfer function of the controller ($k_I/s + k_p$) and the new closed-loop transfer function. Verify that it comes out to:

$$H_{CL}(s) = \frac{\omega(s)}{r(s)} = \frac{(k_I + sk_P)k_1}{s^3 + (a_1 + a_2)s^2 + (a_1a_2 + k_Pk_1)s + k_Ik_1}$$

Now, we'll set the poles somewhat like the critical damping, that is, we'll make all the poles the same, say at -p. That would make:

$$s^3 + (a_1 + a_2)s^2 + (a_1a_2 + k_Pk_1)s + k_Ik_1 = s^3 + 3ps^2 + 3p^2s + p^3$$

Find p, $k_I$, and $k_p$ by setting up 3 equations for the coefficients of the s terms. They should be about 240, 24 and 0.2 respectively.

Under "Control Type", uncheck the "Proportional Controller" and check the "PI Controller". Set the "Feed-forward gain" to 1, and the other gains to what you have just found. Try this new control system with the same speed steps as before. How's the stability look now? How about speed of response? How about the steady-state error? Plot these results.

Do the eraser trick again to see how this system responds to disturbances.

## Conclusion

Check-off and conclude as always. Make sure you compare the various approaches to reducing the steady-state error.