Assignment-PI Controller for velocity

For motor 2, plant transfer function is

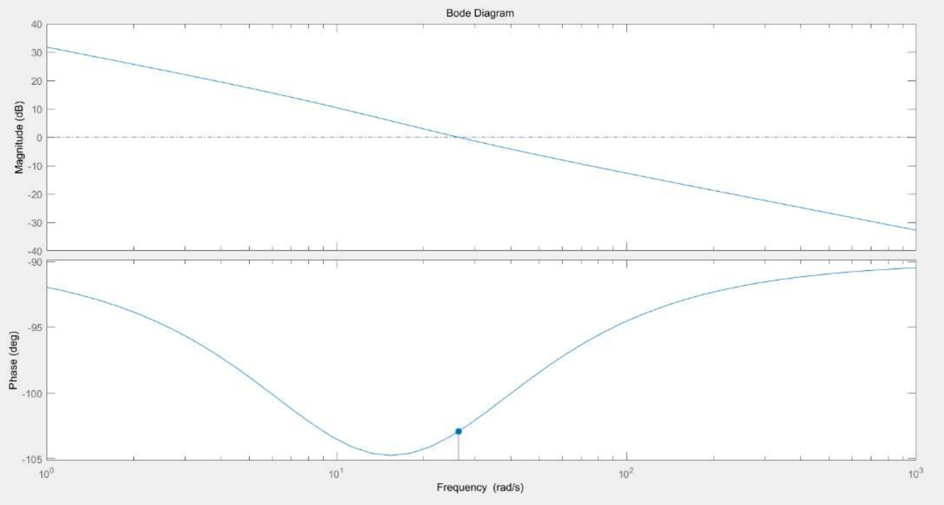
$$g = \frac{0.02305}{s+1\ \ .88}$$

Our sampling frequency is 100 Hz = 628 rad/s, Nyquist frequency is 314 rad/s.

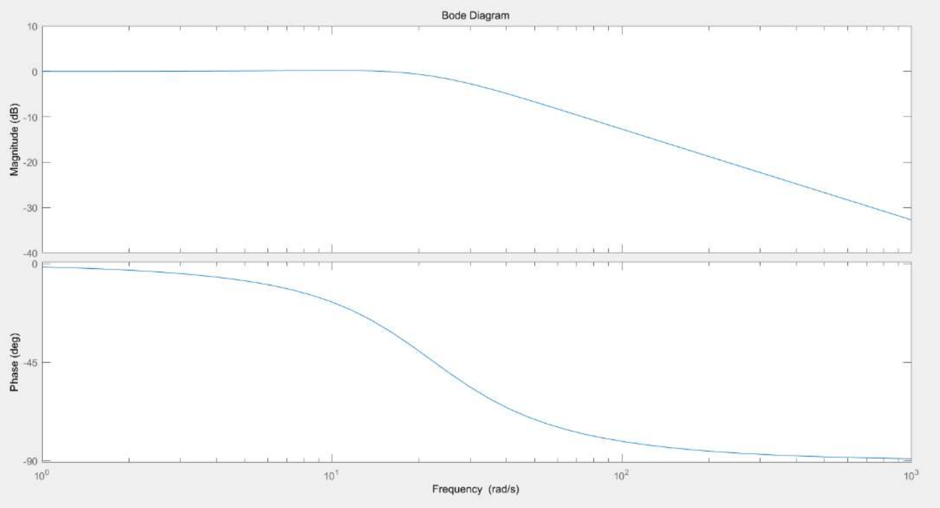
1. First, following the textbook, let set the bandwidth to 628/20 = 31 rad/s. We choose the open-loop crossover frequency as 20 rad/s, because close-loop bandwidth falls in the range of $[\omega_c, 2\omega_c]$. Besides, we put the break point at the open-loop crossover frequency to get a smaller PM (Now it is 77 degrees). It can give us a faster response.
This lead to a PI controller:

$$d = \frac{1000(s+20)}{s}$$

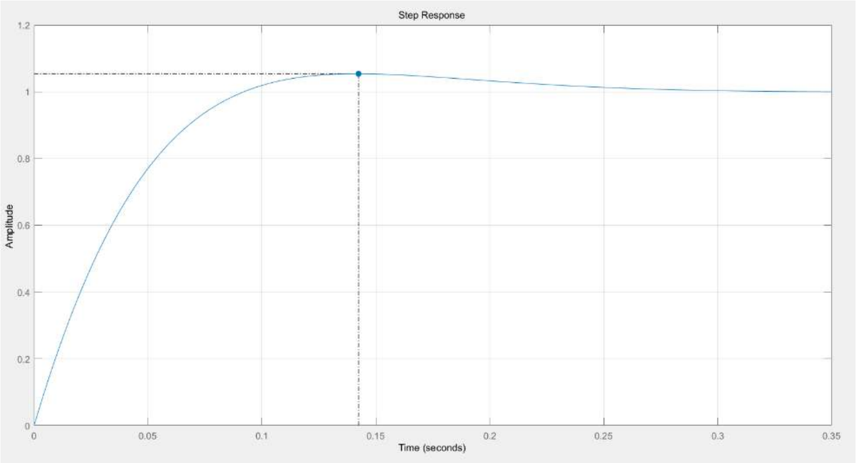
The resulting open-loop bode diagram is shown below: PM = 77.1, ω_c=20 rad/s.



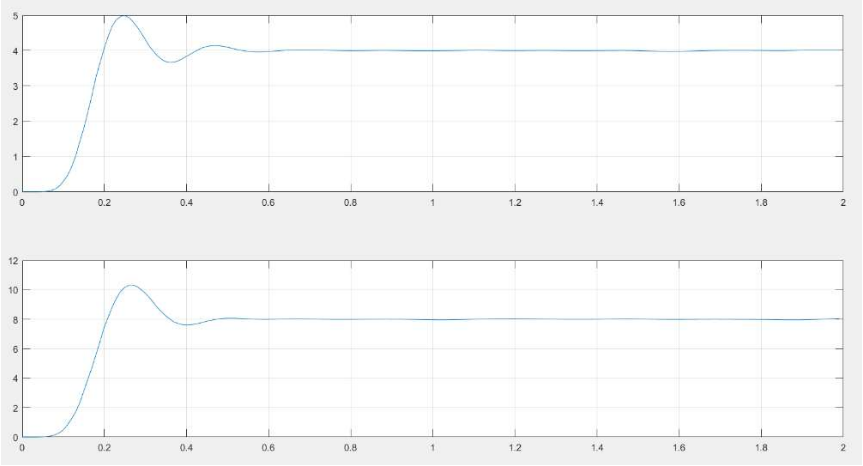
And this is the close-loop bode diagram: bandwidth = 32 rad/s.



Simulated step response: Mp = 5%, settling time about 0.3s.



The test results are shown below. The upper one has reference speed 1 rad/s and the lower one is 2 rad/s. The difference could be explained by the existence of dry friction. Besides, we have neglected the pole of electrical constant, This pole gives us more overshoot than we expect.

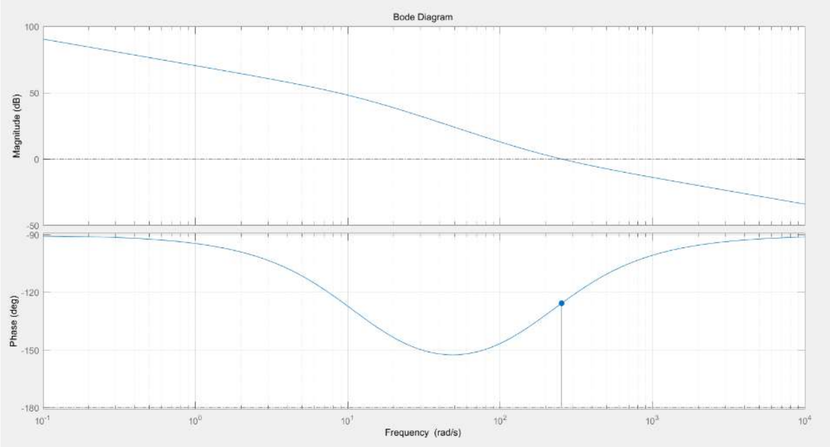


2. Now, let’s try a PI controller that gives the system a close-loop bandwidth of 320 rad/s, which is higher, but very close to, than the Nyquist frequency.

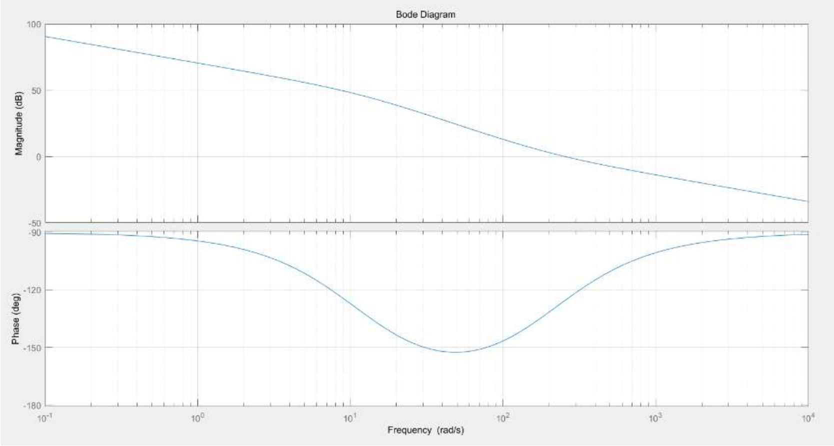
PI controller:

$$d = \frac{8600s+1720000}{s}$$

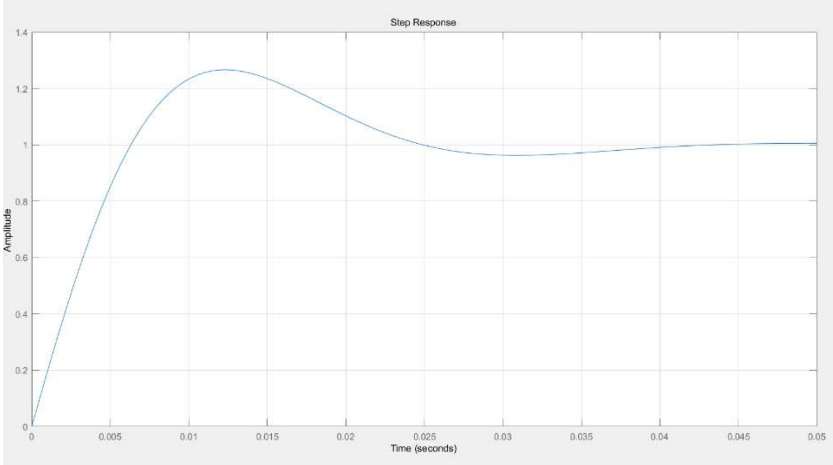
Open-loop bode diagram: PM = 54.3, crossover frequency 253 rad/s



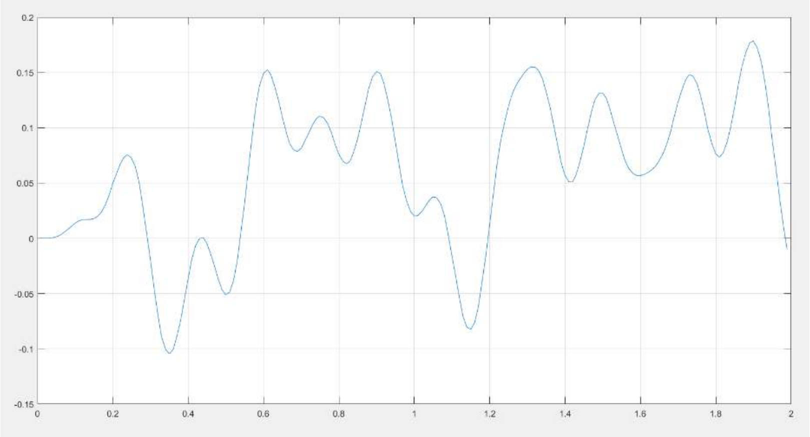
The close-loop bode diagram: bandwidth around 320 rad/s



Simulated step response:
a lot faster than the previous one, due to the much larger bandwidth.

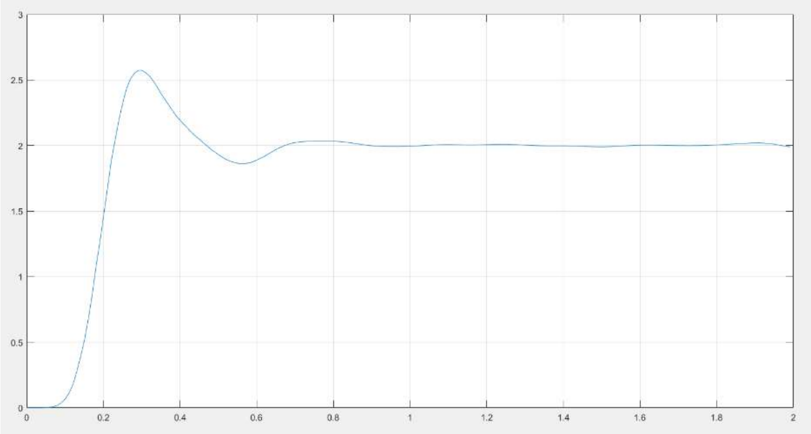


However, the test result is crazy.



On the one hand, the aliasing happened. On the other hand, the open-loop gain is too high which saturated our motor from the beginning.

If we set the close-loop bandwidth to 67 rad/s, the sampling frequency is roughly 10 times larger. The test result shows no improvement, in fact worse, compared with bandwidth=32 rad/s.


3
2.5
2
1.5
1
0.5
0
0
0.2
0.4
0.6
0.8
1
1.2
1.4
1.6
1.8
2