

Chapter 7 HW 1 Solution

Problem 1. First compute the Ψ matrix, which is given by series expansion

$$\Psi = \mathbf{I} + \frac{\mathbf{A}T}{2!} + \frac{\mathbf{A}^2T^2}{3!} + \dots, \quad (1)$$

and when \mathbf{A} and \mathbf{B} and T are substituted we obtain

$$\Psi = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} + \begin{bmatrix} 0 & 0.05 \\ 0 & 0 \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} + \dots [\text{all zeros}] \dots = \begin{bmatrix} 1 & 0.05 \\ 0 & 1 \end{bmatrix} \quad (2)$$

Using Ψ in (7.20) and (7.21) yields

$$\Phi = \begin{bmatrix} 1 & 0.1 \\ 0 & 1 \end{bmatrix}, \quad \Gamma = \begin{bmatrix} 0.005 \\ 0.1 \end{bmatrix} \quad (3)$$

which agrees with the MATLAB `c2d` function (and my notes). Note that the convergence of matrix Ψ was quite rapid. It's not usually quite *that* fast.

Problem 2. We have the armature voltage-controlled DC motor driving a load inertia *via* a gear train.

As before, the state vector is $\mathbf{x} = [i_a \ \theta_m \ \omega_m]^T = [x_1 \ x_2 \ x_3]^T$, and the input is applied voltage $e_a = u$. We are certainly interested in load position (deg), we might also be interested in load velocity (deg/s), and motor current (A) as outputs (for plotting purposes; the *real* output is probably load position only), and the feedthrough matrix \mathbf{D} is zero. Thus the output is $\mathbf{y} = [i_a(\text{A}) \ \theta_l(\text{deg}) \ \omega_l(\text{deg/s})]^T$, and we have

$$\mathbf{A} = \begin{bmatrix} -R_a/L_a & 0 & -K_b/L_a \\ 0 & 0 & 1 \\ K_t/J_t & 0 & -B_t/J_t \end{bmatrix}, \quad \mathbf{B} = \begin{bmatrix} 1/L_a \\ 0 \\ 0 \end{bmatrix}, \quad \mathbf{C} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 180/(n\pi) & 0 \\ 0 & 0 & 180/(n\pi) \end{bmatrix}, \quad \mathbf{D} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}. \quad (4)$$

The inertia and damping terms J_t and B_t represent the “total” inertia of motor armature plus load (remember that the load inertia/damping are *referred* to the motor shaft by dividing by gear ratio n^2).

This motor/load is *uncontrolled*, that is, there is no feedback. There is no way to control the motor's (or load's) position, *etc.* If you give the motor/load some non-zero initial state it will just do whatever—it certainly won't return to zero!

If we design a *state-feedback controller* for this motor/load we can impose closed-loop control on the entire state vector; the motor/load system will then have “good” natural behavior. Later we can think about adding a reference input to control the particular variable of interest (probably load position or velocity).

To design a state-feedback controller, we need to place the poles of the controlled system at reasonable locations. Since the velocity step response had a time constant of near 0.01 second (which corresponds to a pole at $s = -100$), it is reasonable to place all three system poles somewhere near $s = -100$. I decided to place one pole at -100, and the other two as complex conjugates at a radius of 100 and angles of 45° from the negative real axis.

Thus the desired s -plane poles are:

$$s = \underbrace{-100}_{\tau=0.01}, \quad \underbrace{-70 \pm j70}_{\zeta=0.7, \omega_n=100} \quad (5)$$

Mapping these poles to the z -plane yields

$$z = \begin{bmatrix} 0.3679 \\ 0.3798 + j0.3199 \\ 0.3798 - j0.3199 \end{bmatrix} \quad (6)$$

where there is really no reason to keep four decimal places of accuracy!

Before we go further, we need to discretize the plant model; using MATLAB `c2d` the resulting system and input matrices are:

$$\Phi = \begin{bmatrix} -0.0181 & 0 & -0.0055 \\ 0.0195 & 1.0000 & 0.0064 \\ 1.1806 & 0 & 0.3616 \end{bmatrix}, \quad \Gamma = \begin{bmatrix} 0.2538 \\ 0.1573 \\ 28.2421 \end{bmatrix} \quad (7)$$

We can then use MATLAB `place` to find the state feedback gain matrix \mathbf{K} . The result is

```
>> K = place(Phi,Gamma,z_poles)
```

```
K = -0.3512    1.0596    0.0049
```

Since the feedback law is $u = -\mathbf{K}\mathbf{x}$, the control law is actually

$$u = 0.35i_a - 1.06\theta_m - 0.005\omega_m, \quad (8)$$

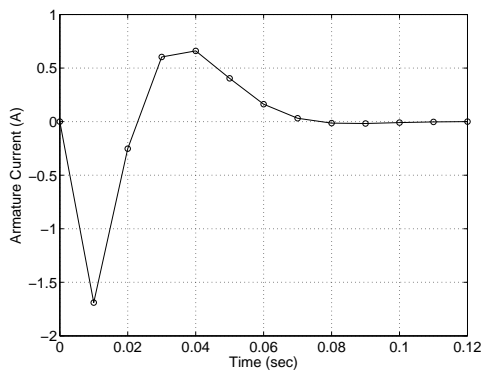
which represents *positive* current feedback, and negative position and velocity feedback.

Response to initial condition. To confirm the behavior of the controlled plant, let's take as an initial state simply a motor "windup" of one revolution (note that this is $1/50$ revolution for the load, or about 7°). Armature current and motor velocity will be left at zero. Thus $\mathbf{x}_0 = [0 \ 2\pi \ 0]^T$. Because of the choice of our \mathbf{C} matrix the output will be armature current, and load position and velocity in degrees.

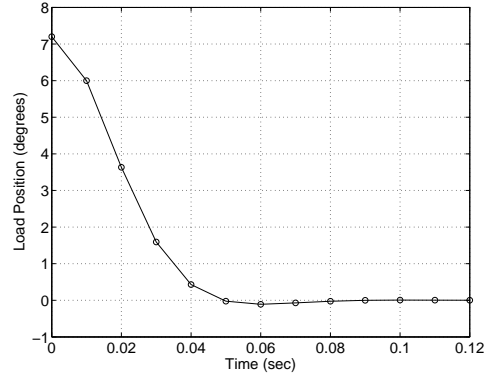
I created the closed-loop system with a zero input matrix, since we don't really have an input yet. And \mathbf{D} is still zero.

```
>> clsys_d = ss(Phi-Gamma*K,zeros(3,1),C,D,T);
```

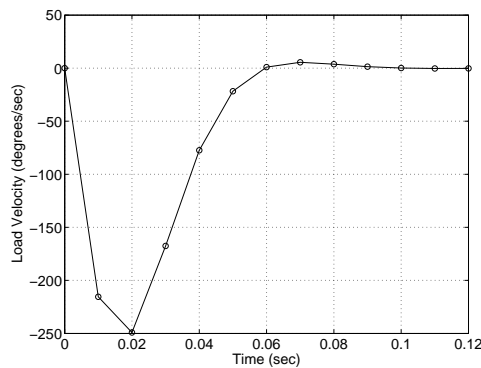
The plots of the three outputs are shown in Figure 1 below:



(a) Current response.



(b) Load position response.



(c) Load velocity response.

Figure 1: Motor/load response to 7° load initial condition.