Homework Assignment #6 (ME-190, Fall 2016)

Due date: Thursday, Oct. 27, 2016, 2:00 pm.

Objectives:

- Compare 1st and 2nd order DC motor models, with and without friction
- Determine what model is a "good enough" approximation to use

Background Information:

The 2nd order transfer function for a DC motor (from voltage input to angular velocity output) is:

$$G(s) = \frac{\omega(s)}{V(s)} = \frac{K_t}{LJs^2 + (JR + Lb)s + (Rb + K_t K_b)}$$

If the inductance is "small" it can be neglected (L=0) and the first order transfer function is obtained:

$$G(s) = \frac{\omega(s)}{V(s)} = \frac{K_t}{JRs + (Rb + K_t K_b)}$$

Where the motor parameters are:

- K_t – torque constant
- K_b – back emf constant
- b – viscous damping coefficient
- R – armature resistance
- J – armature resistance (or combined motor and load inertia if load is attached)
- L – inductance

This lab will examine the effects of the inductance and damping terms. It will also examine the effect of nonlinear coulomb friction (which cannot be included in a transfer function)

$$F_c = sign(\omega) * T_f$$

Where F_c is the total coulomb friction torque – it is a constant magnitude T_f that depends on the direction of rotation $sign(\omega)$. This nonlinear effect will be included in the simulation.

Define Important Motor parameters:

Create an m-file that will have all the important motor parameters – these values may need to be changed for your particular DC motor:

```
% Motor_Parmaeters.m
Rm = 10;      % ohms
Kb = .7;      % Vs/rad
Kt = .7;      % Nm/A
Bm = 0.01;    % Nms/rad (viscous friction)
Lm = 0.004;   % H
Jm = 0.002;   % kgm^2 (combined J)
Tf = 0.02;    % Nm (coulomb friction)
encoder_counts=720;    % number of counts (if using quad encoding)
Vsupply=4.5;           % driver supply voltage (max saturation voltage)
```

Figure 1: Motor parameters stored in script "Motor_Parameters.m"

Simulation Diagram

Create the following simulation diagram that has 3 different DC motor models, a 2nd order, a 1st order and a 1st order with Coulomb/Dry friction. The first two blocks use transfer function blocks, the third is a subsystem you will create as shown below.

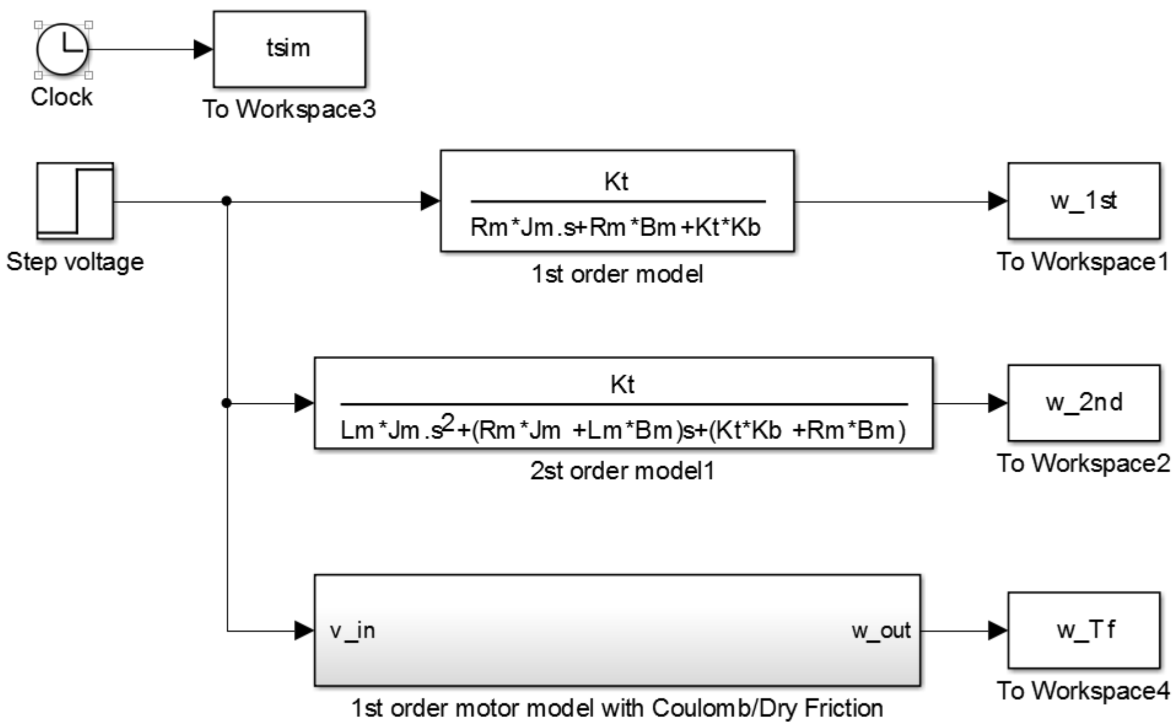


Figure 2: Simulink diagram for 3 different motor models

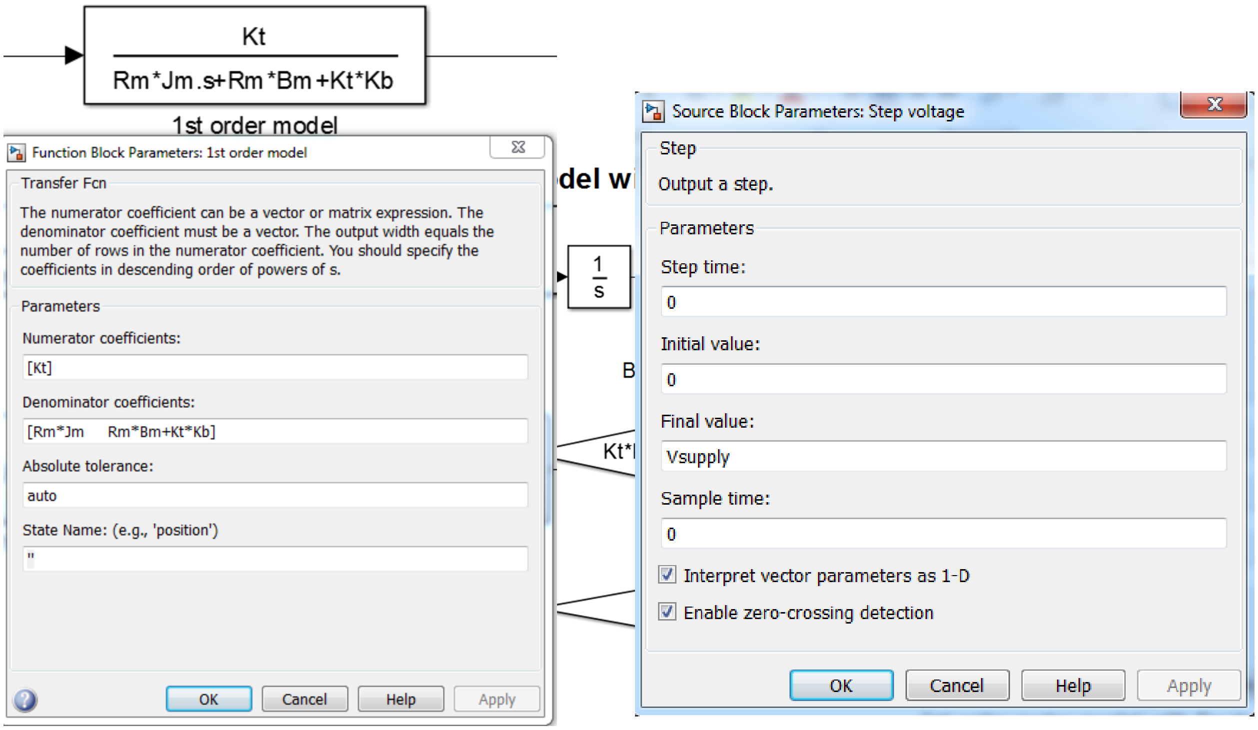
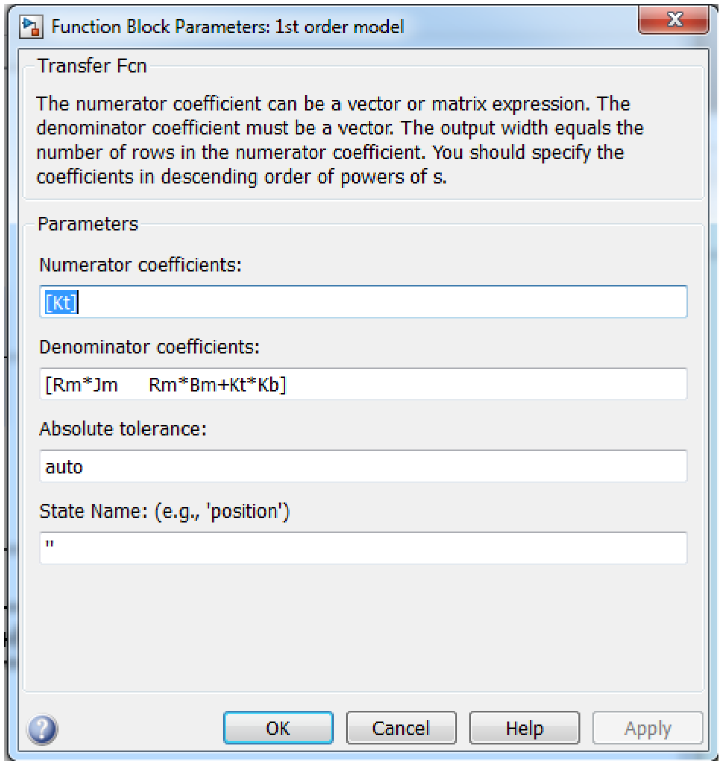


Figure 3: Transfer function and Step block parameters

Enter a transfer function by specifying the coefficients with spaces between them, for example the first order model transfer function is entered:



Simulation Script

Create an m-file that will contain all the important simulation settings, run the simulations and plot the results:

```
clc, clear all, close all

% load motor parameters:
Motor_Parameters;

% simulation parameters
Tsim_final=.3;          % final simulation time in seconds
sim_step_size = .0001; % how often we want data printed from the simulation

% Simulate the model
sim('Motor_Step_Response_Linear_1st_2nd_Coulomb')

% plot the response in RPM
figure(1), hold on
h1=plot(tsim, w_1st*RADSEC2RPM, 'b');
h2=plot(tsim, w_2nd*RADSEC2RPM, 'r');
htf=plot(tsim, w_Tf*RADSEC2RPM, 'g');

% Simulate with no Friction and plot
Bm=0; Tf=0;
sim('Motor_Step_Response_Linear_1st_2nd_Coulomb')
h1_nf=plot(tsim, w_2nd*RADSEC2RPM, 'c--');
h2_nf=plot(tsim, w_Tf*RADSEC2RPM, 'm--');

xlabel('Time (seconds)')
ylabel('angular velocity (RPM) - input 4.5v step')

legend([h1 h2 htf h1_nf h2_nf],...
    {'1st order - viscous only',...
    '2nd order - viscous only',...
    '1st order - coulomb','1st order - no coulomb no viscous',...
    '2st order no viscous'})
```

Figure 5: m-file "Sim_Motor_Response.m" used to simulate and plot motor model simulations

Note: Choose array format for your "To Workspace" blocks, and use the following simulation solver settings:

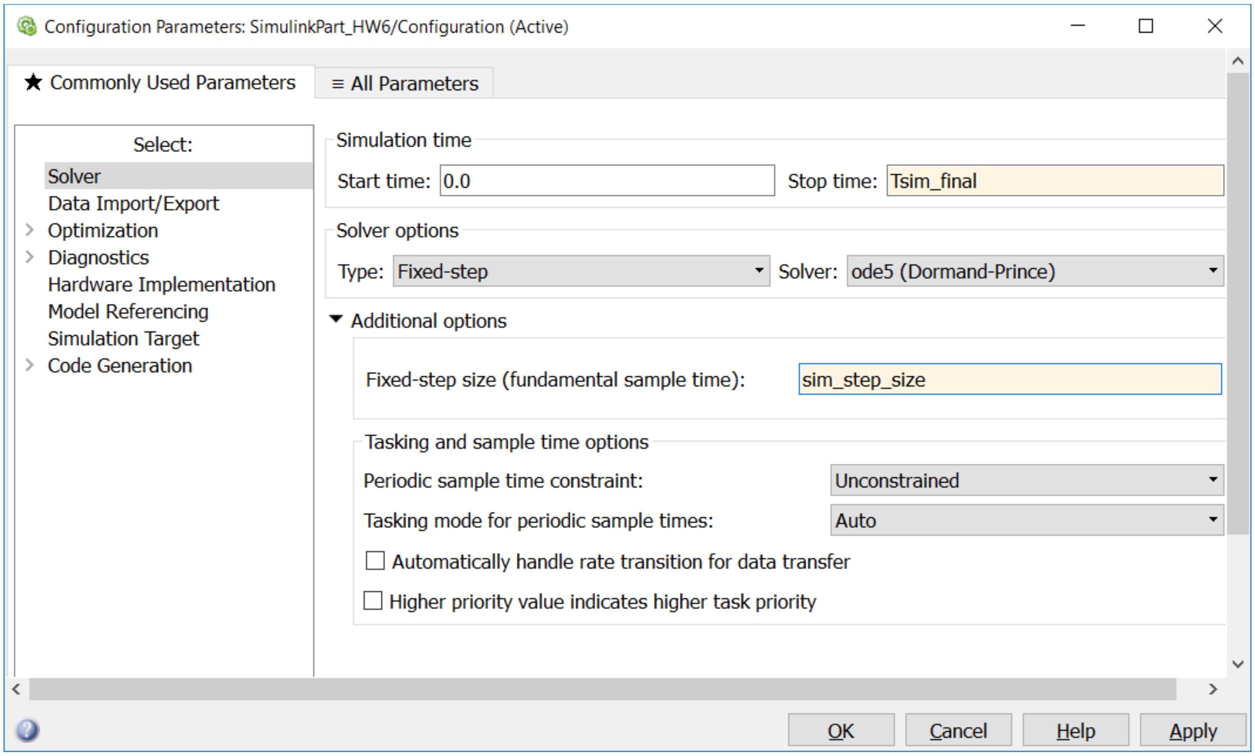


Figure 6: Simulation settings used to solve system response

Run the simulation with and without friction as specified in the m-file and plot the results.

Publish (or generate) a report which includes:

1- The step response plots for the three DC motor models

2- Comparison of the first order system response for the case of having no viscos and no Coulomb friction terms, and an explanation on which term can be ignored without compromising the accuracy.

3- Written Matlab codes and Simulink Models (including the subsystem with friction)