Modeling and Control of Mechatronic Systems

Exercise 1 : Mathematical Modeling and Simulation

Dr Jayantha Katupitiya

Exercise 1

Using Matlab for Simulation

Aim

The aim of this exercise is to learn how to use Matlab scripting language to write programs to simulate a set of state equations. The state equations may represent any system including a set of kinematic equations or dynamic equations. In this exercise the state equations to be used are those given in equation (3.12) that can be found in the set of notes titled 'Topic 3: Mathematical Modeling and Transfer Functions. We have chosen to use two M-files (the files with extension `.m`), one containing the state equations, also called the mathematical model, and the other using the file containing the mathematical model to carry out the simulation. As a convention, we will use the file names as follows. The file containing the model will have names of the form `XXXX_model.m`. The file that carries out the simulation will have the name `XXXX_simulate.m`.

An example 'model' file (`mtrn9211_first_model.m`) is given below:

```
function dy = mtrn9211_first_model(t,y)
% Initialisation
B = 40e-9;
J = 1e-3;
L = 4e-3;
R = 2.482;
n = 2;
ke = 0.0662e0;
kt = 0.0662e0;

% State variables
% [x1 x2]
% x1 - Omega the controlled quantity in rad/s.
% x2 - \dot{Omega} acceleration

% control inputs
% v - Applied voltage v(t)
v = 24;

dy = zeros(2,1);
```

```
    % update state variables (refer to eq (3.12) and fill in the state
    % equations below
    x1 = y(1);
    x2 = y(2);

    dy(1)=x2;
    dy(2)= -(L*B+R*J)/(L*J)*x2-(R*B+n*n*ke*kt)/(L*J)*x1+(n*n*kt/L/J)*v;
```

An example 'simulate' file (`mtrn9211_first_simulate.m`) is given below

```
% This file simulates the model described in mtrn9211_first_model.m

options = odeset('RelTol',1e-4,'AbsTol',[1e-5 1e-5]);
[t,y] = ode45(@mtrn9211_first_model,[0 5],[0 0], options);
figure(1);
plot(t,y(:,1),'b');
%axis equal;
title('Speed plot');
```

1. Simulate the open loop motor model and obtain a plot of speed versus time for:
   (a) A step voltage input applied at time $t = 0$.
   (b) A step voltage input applied at time $t = 3$.
2. Implement a proportional controller and adjust the proportional gain to obtain fastest possible response without overshoot
3. Implement a PD controller and adjust the gains to obtain best possible response.
4. Implement a PID controller and tune it to obtain the best possible response.
5. Change the state equations to enable the implementation of a position controller. Implement a proportional controller to control the position.