

Lab 3 Report

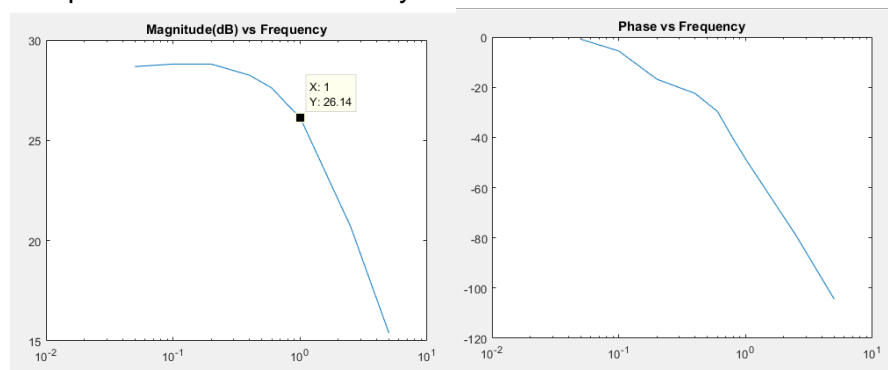
Introduction

The purpose of this lab was to identify the transfer function and the parameters of QNET Allied Motion CL40 Series Coreless DC Motor (model 16705). By using Matlab we interacted with the DC Motor and attempted to identify the model parameters of the DC motor by using the properties of its time and frequency response.

Question 1:

Frequency (Hz)	Ay	Y*	Magnitude	Phase
0.05	27.157434	27.15434	28.67777466	0.8648833606
0.1	27.551020	27.419825	28.80275364	5.593709657
0.2	27.551020	26.370262	28.80275364	16.83498067
0.4	25.845481	23.877551	28.24769238	22.50324326
0.6	24.008746	20.860058	27.60738954	29.67443955
0.8	21.778426	16.530612	26.76052977	40.620702
1.0	20.27389	13.401274	26.13874172	48.62294043
2.5	10.834395	2.050955	20.6960933	79.08804249
5.0	5.885350	-1.471338	15.39544591	104.4775172

4. From the Bode plot it is clear to see the system order is one.



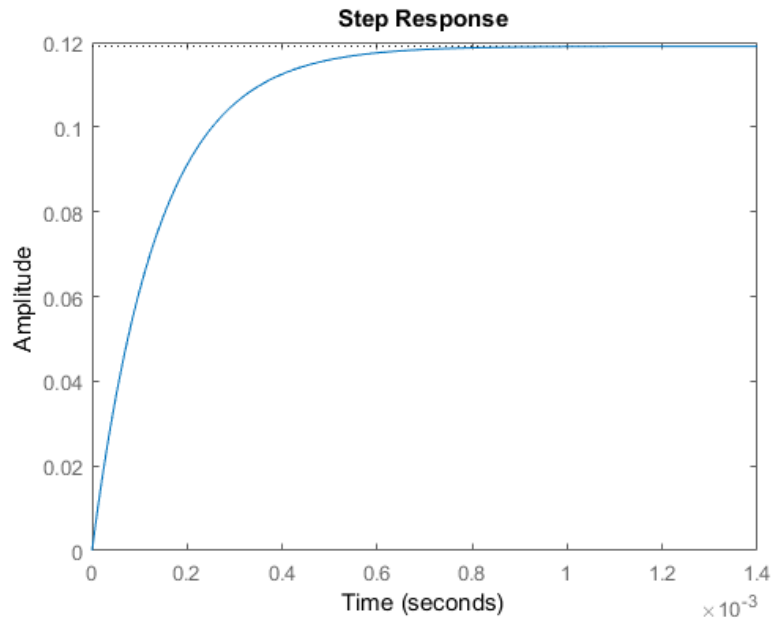
5. The DC gain is the value of the bode plot at low frequencies which is observed to be roughly 28.67 dB.

6. From the bodeplot the 3dB frequency can also be measured to be roughly 1.1 Hz.

Question 2:

Lab 3 Report

1. $H_{in}(s) = 1 / (Ls+R)$



2. The unit step response of the inner loop is shown below:

3. From the plot, the rise time of $H(s)$ is calculated to be $3.034e-04$. Calculating the rise time for the entire system we get

$$t_r = 1.8 / \omega_n$$

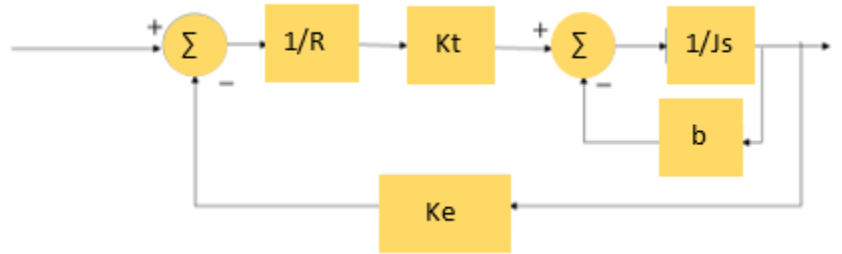
$$t_r = 1.8 / (1.1 * 2 * \pi) = 0.2604s$$

To calculate the order of magnitude we divide the larger number by the smaller number, which gives us: $0.2604 / (3.034e - 04) = 858.2$ this gives us a roughly **three orders of magnitude** difference between rise time of the initial system and the rise time of the whole system.

4. The steady state value of $I(s)$ for step input = $sI(s)$ as s approaches 0
 = $H_{in}(s)$ as s approaches 0
 = $1 / (Ls+R)$ as s approaches 0
 = $1 / R$

Thus, we can replace H_{in} with the equivalent steady-state gain = $1/R$. This result is shown in the simplified block diagram below.

Lab 3 Report

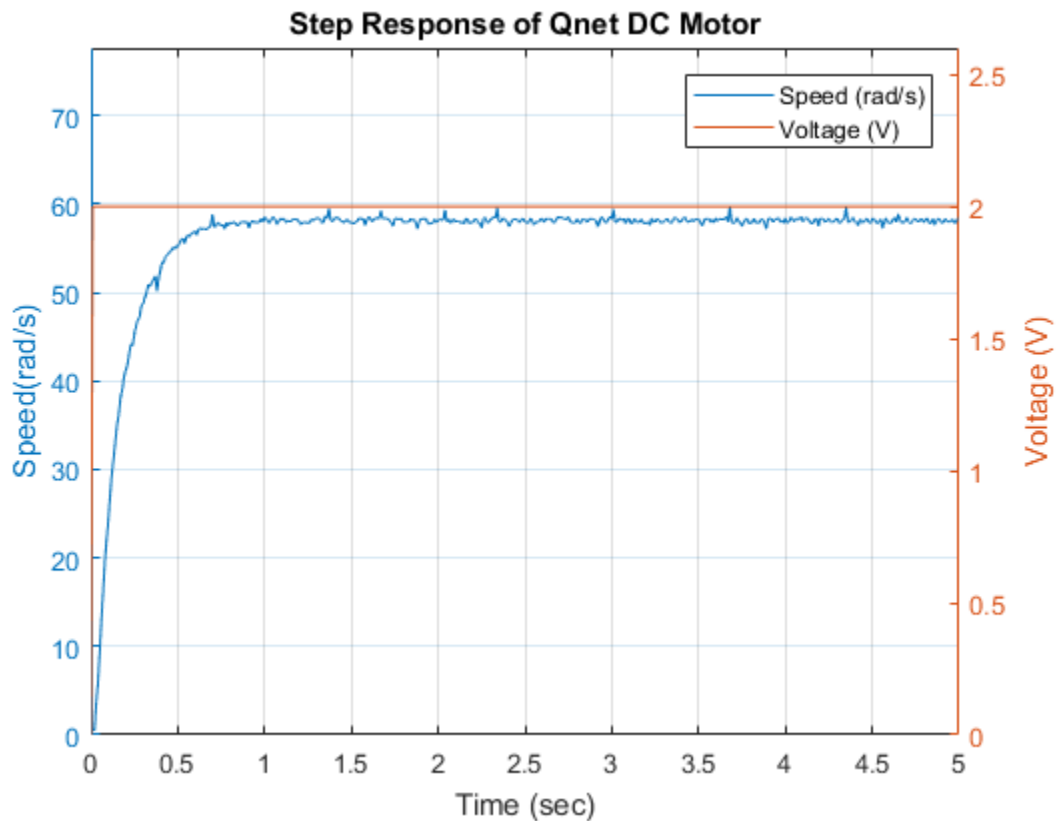


Our value of $(1/R)$ is calculated using Table 1 to be $1/8.4 = 0.119$

5. The reduced transfer function for $\Omega(s)/V(s) = \frac{Kt}{\{R(Js + b) + KtKe\}}$

Question 3:

1. The simulated step response is shown below:



3. Using matlab and the the cursors available for the plot, our value is found to be $\tau = 0.17$.

Using matlab, we were able to calculate the DC gain of the system to be:

DC Gain = $29.0642 = 58.1283 \text{ (rad/s)} / 2V$

Lab 3 Report

4. We are able to solve for the value of b by using the relationship dictated by the steady state response of the mathematical motor model. This result is given by $\Omega(s)/V(s) = \frac{Kt}{(Rb + KtKe)}$. Plugging in value for $Kt = 0.042$, $Ke = 0.03$, $R = 8.4$ and $\Omega(s)/V(s) = 29.062$ V/V. From this we can solve the value of b to be $2.20 \cdot 10^{-5}$ kgm².

Using the rise time and the equation for $T = J / (b + 0.12 * Kt * Ke) = 0.17 = (J)/((2.20 \cdot 10^{-5}) + 0.12(0.042) \cdot (0.03)) = 2.9 \cdot 10^{-5}$ kgm²/s.

Conclusion:

Using the system characteristics, we were able to infer the number of poles, as well as find its general root location. Using the time and frequency response of the system we were able to determine its general transfer function and develop values for using the motors frequency and time responses to output. Each result was measured and plotted using Matlab. So we identified the transfer function and the model parameters of the DC motor by using the properties of its time and frequency response, observed via Matlab.

Appendix:

See matlab code attached below:

```
% -----  
%  
% title      : Lab 3  
%  
% subtitle  : Identification of Qnet DC Motor  
%  
% date      : Week of 09 October, 2017  
%  
% -----  
%  
  
%% Initialization  
% Clear all input and output from the Command Window  
clc  
  
% Many helper functions are provided in the `Interface` sub-directory.  
% We start by adding that directory to Matlab's search path.  
addpath('Interface');  
  
% Create an object handle to interface with Qnet DC Motor  
Motor = QnetDCMotor();  
  
%% 3 - Identification by Frequency Analysis  
  
% Frequency vector  
FreqVect = [0.05, 0.1, 0.2, 0.4, 0.6, 0.8, 1.0, 2.5, 5.0];
```

Lab 3 Report

```
idxFreq = 1; % select the desired index inside the frequency vector

dt = 0.01; % Choose an appropriate sampling time (in seconds)
duration = 30; % Choose an appropriate total duration of
simulation (in seconds)
time = 0:dt:duration; % A vector containing all time samples

% Simulate driving DC motor
Freq = FreqVect(idxFreq); % select frequency from FreqVect
Motor.setSamplingTime(dt);
Motor.reset;
for t = time
    % Generate a cosinwave input at current time
    u = 2 + cos(2*pi*Freq*t);

    % Drive motor for a duration of dt
    Motor.drive(u, t, dt);
end
Motor.off;

%=== Plot example ===%

t = Motor.time;
y = Motor.velocity;
u = Motor.voltage;
delay = 0; % wait for output to reach steady-state.
figure(2)
clf;
yyaxis left
plot(t(t > delay & t <= duration), y(t > delay & t <= duration))
ylabel(['Speed (' Motor.Units '/s)'])
yyaxis right
plot(t(t > delay & t <= duration), u(t > delay & t <= duration))
ylabel('Voltage (V)')
xlim([0,duration])
xlabel('Time (sec)')
legend(['Speed (' Motor.Units '/s)'], 'Voltage (V)')
title(['Input and output vs time at frequency ' num2str(Freq) ' Hz'])
grid on

figure(3) % Lissajous method
clf;
yEllipse = y(t > delay & t <= duration);
uEllipse = u(t > delay & t <= duration);
plot(uEllipse, yEllipse)
xlabel('Voltage (V)')
ylabel(['Speed (' Motor.Units '/s)'])
```

Lab 3 Report

```
title(['XY plot for frequency ' num2str(Freq) ' Hz'])
grid on

data = ginput(4); % click 4 times on the graph to retrieve 4 X,Y values

ay = (data(1,2) - data(2, 2))/2;
ax = (data(3,2) - data(4, 2))/2;

fprintf("Ay: %f\n", ay);
fprintf("Y*: %f\n", ax);
%% 4 - Reduced DC Motor Model %%
s = tf('s');
Hin = 1/((1.16/1000)*s+8.4);
figure(4);
title('Step Response of Hin');
step(Hin);
stepinfo(Hin)
%% 5 - Identification by Step Response Analysis

Motor.reset(); % reset the motor internal variable (including time)
% Drive Motor with a an input voltage for a duration starting from a delay
Motor.drive(2, delay, 5);
Motor.off();

%=== Plot example ===%

t = Motor.time;
y = Motor.velocity;
u = Motor.voltage;
figure(1)
clf;
yyaxis left
plot(t(t>0), y(t>0))
ylim([0 1.3*max(y(t>0))])
ylabel(['Speed(' Motor.Units '/s)'])
yyaxis right
plot(t, u)
ylim([0 1.3*max(u(t>0))])
ylabel('Voltage (V)')
legend(['Speed (' Motor.Units '/s)'], 'Voltage (V)')
grid on
xlim([0 5])
xlabel('Time (sec)')
title('Step Response of Qnet DC Motor')

dcGain = mean(y(t>1 & t<5)) / 2;
```

Lab 3 Report

```
disp(dcGain);
```