
```
% run_model_improvement.m
%
% Runs model improvement code
```

Clear and set all variables

```
clc, clear all, close all
% Variable parameters. This function aims to identify improved parameters

% starting guesses to variable parameters
param_var.mu = 0.254;           % coefficient of friction
param_var.cd = 0.67;           % drag coefficient
param_var.jm = 1.1e-6;         % motor rotor inertia in [Kg*m^2]
param_var.Tau_f = 0;           % motor friction [N*m]

% Fixed parameters
param_fixed.ngear = 4.4;                % gear ratio
param_fixed.wn = 8200*2*pi/60;          % no load motor speed at 100% PWM
    in rad/s (converted from rpm)
param_fixed.trq_stall = 0.17*0.0980665; % motor stall torque at 100% PWM
    in Nm (converted from kg cm)
param_fixed.nut_mass = 3.20e-3;          % {kg} mass of a single nut
param_fixed.bolt_mass = 7.74e-3;         % {kg} mass of a single bolt
param_fixed.fw_mass = 5.53e-2;           % {kg} flywheel mass
param_fixed.hub_mass = 3.84e-3;          % {kg} hub mass
param_fixed.r_disk = 0.057;              % radius of disk [m]
param_fixed.r_dis = 0.0508;              % distance of nut from center[m]
param_fixed.gearbox_fric_factor = 6;      % factor that multiplies coef of
    friction

% Configuration variables
iconfig=1;                               % Index number
config_ar(iconfig).name = ('Config 1 at 50% PWM');
config_ar(iconfig).nut_ar = [1 0 0 0 1 0 0 0];
config_ar(iconfig).pwm = 50;
config_ar(iconfig).exp_data_filename = 'EncoderData_Config11.txt'; % Filename
% next configuration
iconfig = iconfig + 1;
config_ar(iconfig).name = ('Config 1 at 75% PWM');
config_ar(iconfig).nut_ar = [1 0 0 0 1 0 0 0];
config_ar(iconfig).pwm = 75;
config_ar(iconfig).exp_data_filename = 'EncoderData_Config12.txt'; % Filename
% next configuration
iconfig = iconfig + 1;
config_ar(iconfig).name = ('Config 1 at 100% PWM');
config_ar(iconfig).nut_ar = [1 0 0 0 1 0 0 0];
config_ar(iconfig).pwm = 100;
config_ar(iconfig).exp_data_filename = 'EncoderData_Config13.txt'; % Filename
% next configuration
iconfig = iconfig + 1;
config_ar(iconfig).name = ('Config 2 at 50% PWM');
config_ar(iconfig).nut_ar = [1 0 1 0 1 0 1 0];
```

```

config_ar(iconfig).pwm = 50;
config_ar(iconfig).exp_data_filename = 'EncoderData_Config21.txt'; % Filename
% next configuration
iconfig = iconfig + 1;
config_ar(iconfig).name = ('Config 2 at 75% PWM');
config_ar(iconfig).nut_ar = [1 0 1 0 1 0 1 0];
config_ar(iconfig).pwm = 75;
config_ar(iconfig).exp_data_filename = 'EncoderData_Config22.txt'; % Filename
% next configuration
iconfig = iconfig + 1;
config_ar(iconfig).name = ('Config 2 at 100% PWM');
config_ar(iconfig).nut_ar = [1 0 1 0 1 0 1 0];
config_ar(iconfig).pwm = 100;
config_ar(iconfig).exp_data_filename = 'EncoderData_Config23.txt'; % Filename
% next configuration
iconfig = iconfig + 1;
config_ar(iconfig).name = ('Config 3 at 50% PWM');
config_ar(iconfig).nut_ar = [2 2 2 2 2 2 2 2];
config_ar(iconfig).pwm = 50;
config_ar(iconfig).exp_data_filename = 'EncoderData_Config31.txt'; % Filename
% next configuration
iconfig = iconfig + 1;
config_ar(iconfig).name = ('Config 3 at 75% PWM');
config_ar(iconfig).nut_ar = [2 2 2 2 2 2 2 2];
config_ar(iconfig).pwm = 75;
config_ar(iconfig).exp_data_filename = 'EncoderData_Config32.txt'; % Filename
% next configuration
iconfig = iconfig + 1;
config_ar(iconfig).name = ('Config 3 at 100% PWM');
config_ar(iconfig).nut_ar = [2 2 2 2 2 2 2 2];
config_ar(iconfig).pwm = 100;
config_ar(iconfig).exp_data_filename = 'EncoderData_Config33.txt'; % Filename
% next configuration
iconfig = iconfig + 1;
config_ar(iconfig).name = ('Config 4 at 50% PWM');
config_ar(iconfig).nut_ar = [3 0 3 0 3 0 3 0];
config_ar(iconfig).pwm = 50;
config_ar(iconfig).exp_data_filename = 'EncoderData_Config41.txt'; % Filename
% next configuration
iconfig = iconfig + 1;
config_ar(iconfig).name = ('Config 4 at 75% PWM');
config_ar(iconfig).nut_ar = [3 0 3 0 3 0 3 0];
config_ar(iconfig).pwm = 75;
config_ar(iconfig).exp_data_filename = 'EncoderData_Config42.txt'; % Filename
% next configuration
iconfig = iconfig + 1;
config_ar(iconfig).name = ('Config 4 at 100% PWM');
config_ar(iconfig).nut_ar = [3 0 3 0 3 0 3 0];
config_ar(iconfig).pwm = 100;
config_ar(iconfig).exp_data_filename = 'EncoderData_Config43.txt'; % Filename
% next configuration
iconfig = iconfig + 1;
config_ar(iconfig).name = ('Config 5 at 50% PWM');
config_ar(iconfig).nut_ar = [3 0 0 0 3 0 0 0];

```

```

config_ar(iconfig).pwm = 50;
config_ar(iconfig).exp_data_filename = 'EncoderData_Config51.txt'; % Filename
% next configuration
iconfig = iconfig + 1;
config_ar(iconfig).name = ('Config 5 at 75% PWM');
config_ar(iconfig).nut_ar = [3 0 0 0 3 0 0 0];
config_ar(iconfig).pwm = 75;
config_ar(iconfig).exp_data_filename = 'EncoderData_Config52.txt'; % Filename
% next configuration
iconfig = iconfig + 1;
config_ar(iconfig).name = ('Config 5 at 100% PWM');
config_ar(iconfig).nut_ar = [3 0 0 0 3 0 0 0];
config_ar(iconfig).pwm = 100;
config_ar(iconfig).exp_data_filename = 'EncoderData_Config53.txt'; % Filename
% next configuration
iconfig = iconfig + 1;
config_ar(iconfig).name = ('Config 6 at 50% PWM');
config_ar(iconfig).nut_ar = [1 1 1 1 1 1 1 1];
config_ar(iconfig).pwm = 50;
config_ar(iconfig).exp_data_filename = 'EncoderData_Config61.txt'; % Filename
% next configuration
iconfig = iconfig + 1;
config_ar(iconfig).name = ('Config 6 at 75% PWM');
config_ar(iconfig).nut_ar = [1 1 1 1 1 1 1 1];
config_ar(iconfig).pwm = 75;
config_ar(iconfig).exp_data_filename = 'EncoderData_Config62.txt'; % Filename
% next configuration
iconfig = iconfig + 1;
config_ar(iconfig).name = ('Config 6 at 100% PWM');
config_ar(iconfig).nut_ar = [1 1 1 1 1 1 1 1];
config_ar(iconfig).pwm = 100;
config_ar(iconfig).exp_data_filename = 'EncoderData_Config63.txt'; % Filename
% set total configurations
nconfig = iconfig;

```

Loop to find optimal coefeciant of friction, param_var.mu

The assignment will required nested loops to find optimzal values for mu, Jm, and Cd

```

mu_low_opt = 0.24;
mu_high_opt = 0.26;
mu_step_opt = 0.001;
mu_vec = mu_low_opt:mu_step_opt:mu_high_opt;
mu_len = length(mu_vec);

% cd_low_opt = 0.66;
% cd_high_opt = 0.68;
% cd_step_opt = 0.005;
% cd_vec = cd_low_opt: cd_high_opt: cd_step_opt;
% cd_len = length(cd_vec);

```

```

% jm_low_opt = 1e-6;
% jm_high_opt = 5e-6;
% jm_step_opt = 1e-7;
% jm_vec = jm_low_opt: jm_high_opt: jm_step_opt;
% jm_len = length(jm_vec);

tic
iopt = 1; % mu optimization counter
% iopt2 = 1; % Cd optimization counter

for mu = mu_low_opt:mu_step_opt:mu_high_opt
%     disp([Cd mu])
    disp(mu)
    param_var.mu = mu;
    iconfig = 1;
    for iconfig = 1:nconfig
        config = config_ar(iconfig);
        compare_sim_exp % run and compare experimental to simulation
    end
    err_metric_mean_opt_ar(iopt) = mean(err_metric_ar);
    mu_opt_ar(iopt) = mu;

    iopt = iopt + 1;
end
toc

% min_err_metric_mean = min(err_metric_mean_opt_ar(err_metric_mean_opt_ar>0));
%
% [r, c] = find(err_metric_mean_opt_ar == min_err_metric_mean, 1, 'last');

% fprintf('Most Optimal case corresponds to: \n');
% fprintf('Cd of %.4f \n', Cd_vec(r));
% fprintf('Mu of %.4f \n \n', mu_vec(c-mu_len*(r-1)));

figure(4)
plot(mu_opt_ar,err_metric_mean_opt_ar)
xlabel('Coefficient of Friction - \mu');
ylabel('Mean Error Metric');
title('Error Metric as a Function of Friction Coefficient');

% output_table. Note use of (:) converts row vectors to column vectors
output_table = [tr_exp_ar(:) wterm_exp_ar(:) tr_sim_ar(:) wterm_sim_ar(:)
    tr_err_ar(:) wterm_err_ar(:) err_metric_ar(:)];
writematrix(output_table,'Output Table.csv')

0.2400

0.2410

0.2420

0.2430

0.2440

```

0.2450

0.2460

0.2470

0.2480

0.2490

0.2500

0.2510

0.2520

0.2530

0.2540

0.2550

0.2560

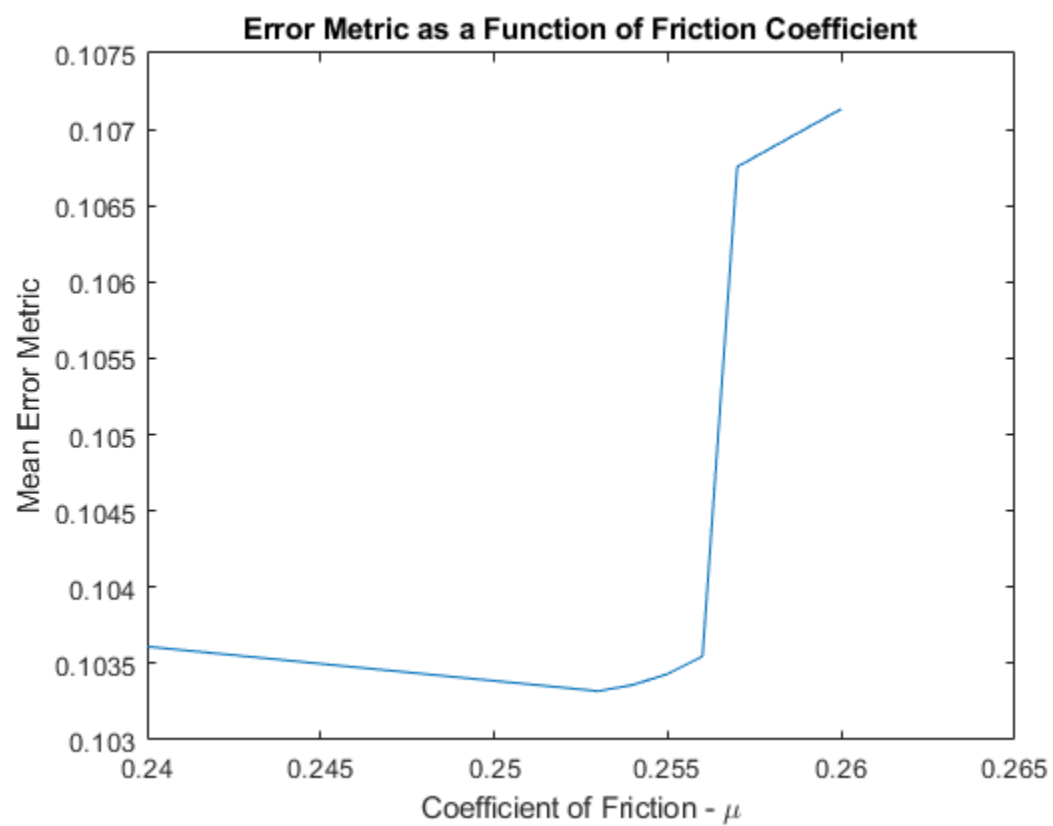
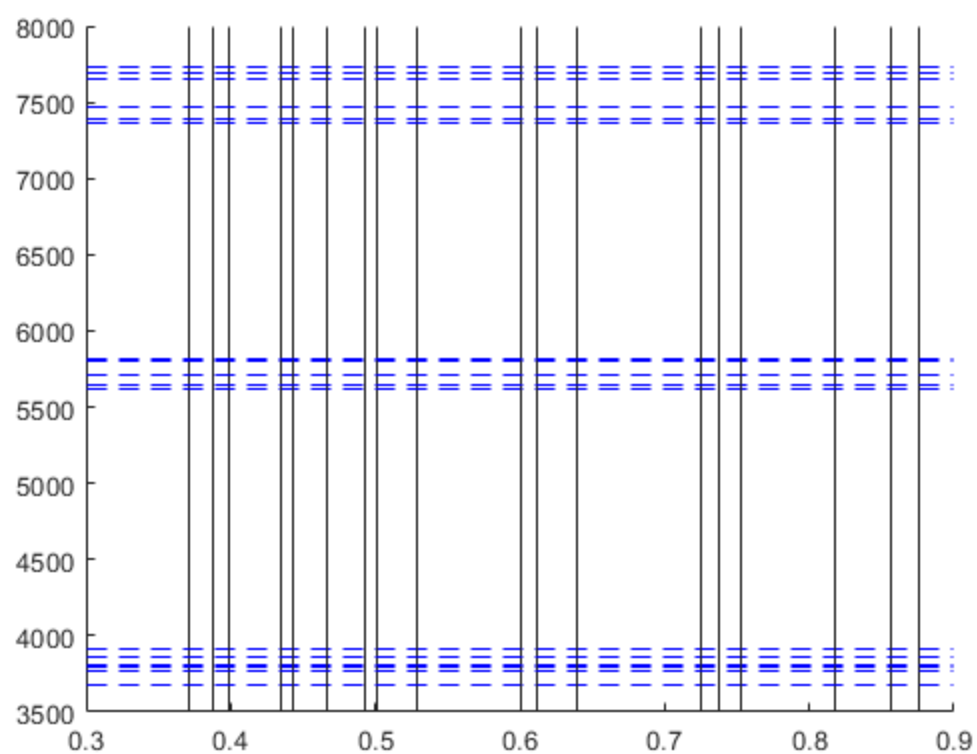
0.2570

0.2580

0.2590

0.2600

Elapsed time is 50.628774 seconds.



Published with MATLAB® R2021b