University of Colorado - Boulder

ASEN 3200 Attitude and Orbital Mechanics

Lab 3 - Section 013

Authors:
Alan Jaffe
Hayden Gebhardt
Michael Lauver Lopez

Professor:
Dr. Casey Heidrich

College of Engineering & Applied Science
UNIVERSITY OF COLORADO BOULDER

Submitted: February 9, 2026

Contents

I. Abstract

This experiment focuses on the design and implementation of active control systems to cause a spacecraft model to follow a reference trajectory. The reference trajectory is defined by one degree of angular freedom. The closed loop tracking behavior is achieved by designing a feedback controller with proportional and derivative gains to achieve a desired system response, with respect to the spacecraft's moment of inertia and rigid body properties.

II. Preliminary questions

1. Description of Spacecraft

The moment of inertia can be calculated using τ = I α, τ being the applied torque, and α being the angular rate of the spacecraft in rad/s with respect to time. Experimentally, we can apply a torque to the spacecraft perpendicular to its axis of free motion, and capture data for its angular rate at the time we apply the torque. Using this, we can easily calculate our value for the moment of inertia by computing I = τ α.

2. Block Diagram

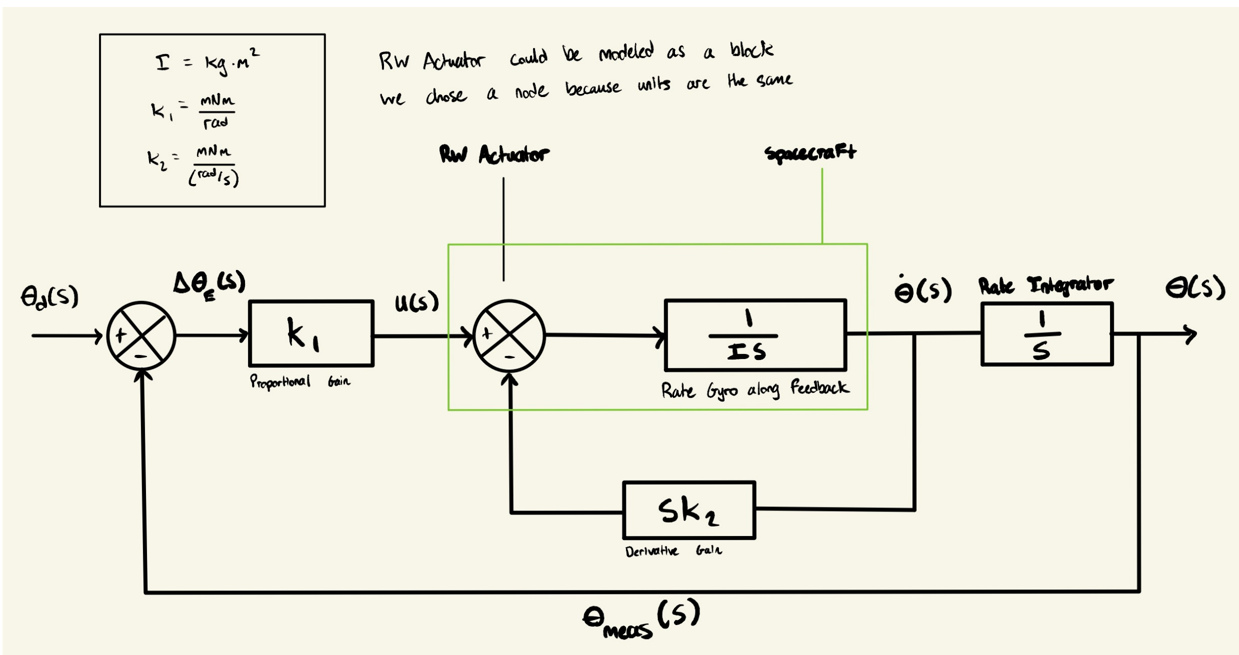

Fig. 1 Single DOF Control Block Diagram

3. Poles and gains of closed loop system

Using our given values, we can calculate the damping coefficient and the natural frequency. We can calculate our damping coefficient using our max overshoot equation, which is given by:

$$M_p = exp(\frac{\zeta}{\sqrt{1-(\zeta)^2}}\pi) \tag{1}$$

We can re-arrange to solve for ζ, which provides us with:

$$\zeta = \frac{-ln(0.1)}{\sqrt{(\pi)^2 + ln(0.1)^2}} \tag{2}$$

Evaluating this equation gives us a damping ratio of ζ = 0.591.

Next, we can evaluate the natural frequency by using the percent overshoot equation, which has the form of:

$$P = exp(-\zeta w_n t_s) \tag{3}$$

Re-arranging to solve for w_n, we get:

$$w_n = \frac{ln(0.1)}{\zeta t_s} \tag{4}$$

Evaluating, we get a natural frequency of w_n = 3.379 rad/s.

Subsequently, if we evaluate our transfer function from the above block diagram, we get that our damping coefficients are:

$$I w_n^2 = K_1 = 11.4461 I \tag{5}$$

$$2\zeta w_n I = K_2 = 4I \tag{6}$$

III. Experiment and Analysis

A. Inertia

Listed below are our average values of the moment of inertia for each trial.

- Case 1: I = 9.4687 [kgm^2]
- Case 2: I = 8.3246 [kgm^2]
- Case 3: I = 7.5829 [kgm^2]
- Case 4: I = 7.8828 [kgm^2]

- Average I: 8.2998 [kgm^2]

These values were derived in MATLAB by utilizing the known equation for the torque, $T = [I]\alpha$ and solving for I. Each case represents the inertia for a particular amount of torque and corresponding angular acceleration, represented as the slope of the angular velocity data. Case 1 is 4 mNm of torque, and Case 2 is 6 mNm, of torque, up to 10 mNm of torque. It is important to note that each trial is simulated for the same amount of time. The only difference between them is the inputted value of torque. We see from our values that the higher the torque, the higher the magnitude of inertia. However, the value for inertia seems to stabilize towards 8.2998 kgm^2. This value is positive because the reaction wheel angular velocity convention is negative for our dataset. Taking into account the maximum torque of the spacecraft to be 20 mNm, it is possible that these values are stabilizing as such since the spacecraft can only handle so much load.

Sources of error could range between human and mechanical errors. Depending on how the data is parsed and collected, we could face some human error with slight deviations in results. Additionally, the machine has an inherent error with things such as mechanical friction.

B. Control Design

Our closed-loop step response needed to have a 5% settling time of less than 1.5s, with an overshoot of less than 10%. From their respective equations, we found that: $\zeta = 0.591$, $\omega_n = 3.379\frac{rad}{s}$. Our proportional and derivative gains were calculated by solving the denominator of the closed-loop transfer function, with respect to the moment of inertia. Once these values were found, we set the denominator equal to zero and solved for the s-values, which are the poles.

The poles of our system are: { -2.0000 + 2.7288i, -2.0000 - 2.7288i}. Both real components of our poles are negative, therefore we can assume the spacecraft is stable when influenced by our closed-loop system. The analytical step and impulse responses of our closed-loop system can be seen below:

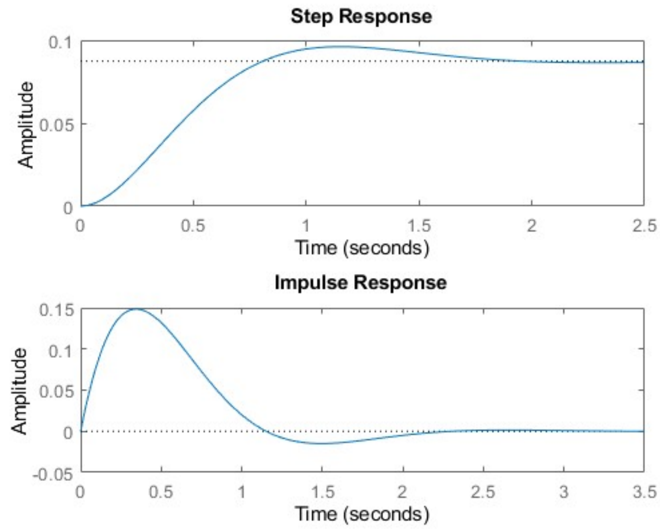

Fig. 2 Analytical Step and Impulse Response

C. Control Results

The experimental results from K1 = 95 and K2 = 33.1667 are shown below. We were given a reference angle of 0.5 rad, shown by both the horizontal line and the reference square wave. The integrated gyro output is referring to our measured square waveform. This measured waveform is the behavior of the spacecraft when provided the aforementioned gains and reference values.

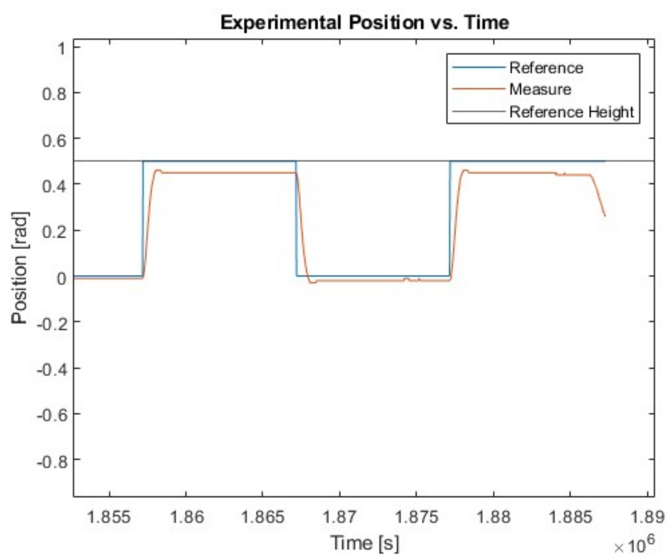

Fig. 3 Experimental Position vs. Time Response

When referencing this plot to our step reference, we see that the measured vs reference position data follows closely to our analytical step response. We see a relative undershoot when we reach our first reference angle, and immediately see an overshoot shortly after we return to an angle of zero, while subsequently reaching just short of zero when our measured data settles. Overall, we do see similar behavior when referenced to our analytical data. Within the short period of overshoot, we reach an overshoot of roughly 2%, which falls within our acceptable range.

D. Control Analysis/Comparison

By inspection, we can see that the settling time of the step response is 1.24s when the response reaches 5% of the steady-state value, which is less than the design maximum of 1.5s. The maximum overshoot relative to the response-steady state is 2.22%, which is less than the design maximum of 10%.

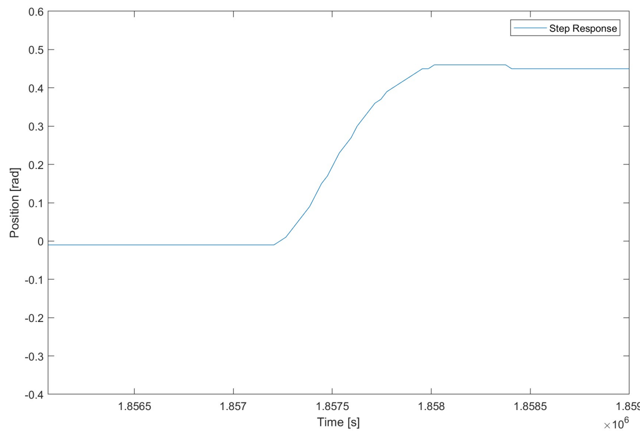

Fig. 4 Reaction Wheel Position / System Step Response

E. Disturbance Response

Varying the proportional gain within the experiment, we notice a trend when increasing or decreasing the value between 0 and 200. At a value of 50, we saw a relatively fast response to the disturbance, with the rotator returning back to zero within a couple of seconds. Increasing K1 provides us with a higher time to decay to zero, resulting in a longer time of oscillation.

If we provide a derivative gain between 0 and 30, we see that our motion ceases a linear motion of oscillation towards zero, and now behaves quadratically, or rather exponentially, towards zero. With the addition of the derivative gain in addition to a linear gain, we see a rapid decrease in oscillation time, with the amplitude of each oscillation exponentially decreasing with time. We know to expect this response since an increase of K1 increases the relative error or "stiffness" our system experiences, while K2 represents the damping of our system. Varying values for K1 and K2, we can see a trend in different system conditions, such as being critically damped (determined from our system) or underdamped (K2 equal to zero).

F. PID vs. PD empirical comparison

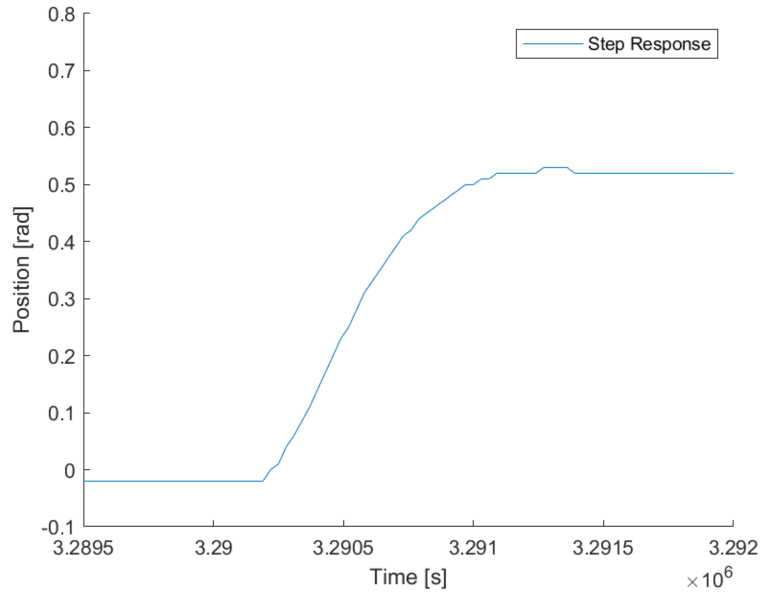

Fig. 5 Analytical Step and Impulse Response with Integral Gain

By inspection, the settling time to reach within 5% of the settling time is 1.2 seconds, which is also below our max design requirement of 1.5 seconds. Our max overshoot relative to the steady state is 1.92%, also less than the design max of 10%. Based on this, we can see that the integral gain removes some long-term errors. In comparison to the PD controller, we have a shorter setting time and a smaller max overshoot, which shows that the integral gain does in fact reduce the systems error relative to the steady state.

IV. Conclusion and Recommendations

This experiment focused on the design and implementation of active control systems with the intention of causing a spacecraft model to follow a reference trajectory. The closed loop tracking behavior was achieved by designing a feedback controller with gains that we calculated by determining the spacecraft's experimental moment of inertia. The proportional and derivative gains were achieved by tailoring the system's closed loop transfer function to the required damping coefficient and natural frequency we desired in our reaction response. In synthesis, we achieved a new depth of knowledge of generating methods and models that allow spacecraft to autonomously perform attitude corrections.

V. Appendix

Participation Report

Name	Contribution Factor	Signature
Hayden Gebhardt	Code and Report	HG
Alan Jaffe	Code and Report	AJ
Michael Lauver Lopez	Code and Report	MLL

We would like to acknowledge the TAs and the Orbital Lab Teaching Team that helped us answering conceptual questions and any misunderstandings we had regarding the lab.

References

The initial version of this lab was created by Prof Dale Lawrence. Modifications have been made by P. Axelrad, H. Schaub, D. Scheeres, Bobby Hodgkinson, Trudy Schwartz.

A. MATLAB Code

```
% Hayden Gebhardt, Alan Jaffe, Michael Lauver Lopez
% ASEN 3200 - Lab 3

%% Housekeeping

clc; clf;
clear all

%1 rpm -> 0.1047 rad/s
%I = L/alpha

data_4 = load("5sec4torque_Lab3GROUP17");
data_4 = data_4(10:end, :);
data_6 = load("5sec6torque_Lab3GROUP17");
data_6 = data_6(10:end, :);
data_8 = load("5sec8torque_Lab3GROUP17");
data_8 = data_8(10:end, :);
data_10 = load("5sec10torque_Lab3GROUP17");
data_10 = data_10(10:end, :);

time1 = data_4(:, 1) / 1000;
time2 = data_6(:, 1) / 1000;
time3 = data_8(:, 1) / 1000;
time4 = data_10(:, 1) / 1000;

%% torque = 4 mNm

t_4 = data_4(:, 2); % mNm
omega_4 = data_4(:, 3) * 0.1047; % RPM -> rad/s
line_4 = polyfit(time1, omega_4, 1);
alpha_4 = line_4(1);

moi1 = t_4/alpha_4;
moi1(any(isinf(moi1),2),:) = [];
moi1final = rmmissing(moi1);
MOI1 = mean(moi1final);

%% torque = 6 mNm

t_6 = data_6(:, 2); % mNm
omega_6 = data_6(:, 3) * 0.1047; % RPM -> rad/s
line_6 = polyfit(time2, omega_6, 1);
alpha_6 = line_6(1);

moi2 = t_6/alpha_6;
moi2(any(isinf(moi2),2),:) = [];
moi2final = rmmissing(moi2);
MOI2 = mean(moi2final);
```

```
%% torque = 8 mNm

t_8 = data_8(:, 2); % mNm
omega_8 = data_8(:, 3) * 0.1047; % RPM -> rad/s
line_8 = polyfit(time3, omega_8, 1);
alpha_8 = line_8(1);

moi3 = t_8/alpha_8;
moi3(any(isinf(moi3),2),:) = [];
moi3final = rmmissing(moi3);
MOI3 = mean(moi3final);

%% torque = 10 mNm

t_10 = data_10(:, 2); % mNm
omega_10 = data_10(:, 3) * 0.1047; % RPM -> rad/s
line_10 = polyfit(time4, omega_10, 1);
alpha_10 = line_10(1);

moi4 = t_10/alpha_10;
moi4(any(isinf(moi4),2),:) = [];
moi4final = rmmissing(moi4);
MOI4 = mean(moi4final);

%% Average MOI

MOI_sc = -(MOI1 + MOI2 + MOI3 + MOI4) / 4

%% Gain calculations

k1 = 11.4461 * MOI_sc
k2 = 4 * MOI_sc

%%

%% Closed-loop response

sys = tf(1, [1 (k2/MOI_sc) (k1/MOI_sc)])

subplot(2,1,1)
step(sys)
subplot(2,1,2)
impulse(sys)

pole(sys)
h = pzplot(sys);

%% Control Plots

clc
clear
close

```

```matlab
%1 rpm -> 0.1047 rad/s
%I = L/W

%% torque = 4 mNm

dataControl = load("K1_95_K2_33control20sLABTASK4");

time = dataControl(:,1); %ms
refpos = dataControl(:,2); %rad
measpos = dataControl(:,3); %rad
current = dataControl(:,4); %Amp

figure()
plot(time,refpos)
hold on
plot(time,measpos)
legend('Reference','Measured')
axis([1856065.000 1887290.000 -1 1])
title('')
xlabel('Time [s]')
ylabel('Position [rad]')

figure()
plot(time, measpos)
legend('Step Response')
axis([1856065.000 1859000.000 -0.4 0.6])
title('')
xlabel('Time [s]')
ylabel('Position [rad]')
```