Examination Paper for Foreign Graduates

Course: MATLAB Programming

Date: 21/12/2021

Title of the Report: MATLAB Programming Final Project

Name: Brayan Antonio Espinoza Garcia

Student No. : LS2125204

School: International School

Category: ▣ Master Program □ Doctoral Program

Nationality: Peruvian

Specialty: Microsatellite Technology

Compulsory/Elective: Compulsory

Score: __________

BEIHANG UNIVERSITY

MATLAB Programming final Project: CubeSat Attitude Control evaluator

line 1: 1st Brayan Antonio Espinoza García 潘诺扎 (LS2125204)
line 2: MATLAB programming
(Microsatellite technology MATA 2021)
line 3: *Beihang University*
line 4: Beijing, China
line 5: bespinozag@unsa.edu.pe

***Abstract*— Attitude Control is an essential part of the "Attitude Determination Control Subsystem" (ADCS) due to its critical impact on the performances of communication antennas, solar panels, and payloads. However, sometimes tuning control algorithms and choosing the best controller is annoying due to the many simulations needed to be carried on. This project presents an easy-to-use GUIDE capable of simulating two three-axis control algorithms. Also, it will measure four performance parameters to make an objective comparison between these algorithms.**

Keywords—MATLAB, Attitude Control, Spacecraft Kinematics and Dynamics.

I. INTRODUCTION

Attitude Control is an essential part of the "Attitude Determination and Control Subsystem" (ADCS). This subsystem is in charge of establishing and controlling the Satellite orientation. Attitude is a critical part of satellites because it affects the communication antennas, solar panels, and payloads [1].

Usually, there are many types of attitude controllers. However, we will focus on the three-axis stabilized Satellite. In this way, many authors propose many control algorithms to accomplish the tracking problem [2]. Most of these controllers were rigorously studied and simulated but choosing the best control algorithm with the best characteristics is usually challenging. The principal objective of this project is to build an easy-to-use GUIDE to simulate and compare two control algorithms in different scenarios with the lower effort. Also, the GUIDE must measure four performance parameters to make an objective comparison and evaluation.

The following report is divided as follows: Section II presents the theoretical framework and mathematical equations used in this project. In Section III we explain the design procedure and each of the parts developed in this project. In section IV we present several tests and results obtained from the developed GUIDE. Finally, in section V the conclusions of the present project are drawn.

II. THEORETICAL FRAMEWORK AND PROJECT DESIGN

This project performs a CubeSat attitude control simulation for many control algorithms. Figure 1 shows the block diagram implemented in this project to perform attitude simulations. [1].

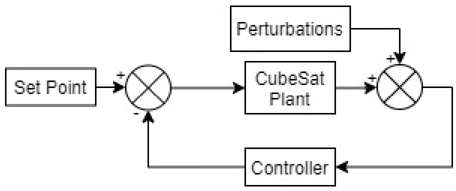

Figure 1: Simulation block diagram.

A. CubeSat Plant

The CubeSat attitude plant is modeled like a rigid body and is composed of (1), (2), and (3). [1], [2]

$$\dot{q} = \frac{1}{2}\Xi(q) \quad (1)$$

$$\dot{\omega} = I^{-1}[Td + U - \omega \times I\omega] \quad (2)$$

Where:

$$\Xi(q) = \begin{bmatrix} q_4 & -q_3 & q_2 \\ q_3 & q_4 & -q_1 \\ -q_2 & q_1 & q_4 \\ -q_1 & -q_2 & -q_3 \end{bmatrix} \quad (3)$$

q: Is a quaternion in the form [q1, q2, q3, q4]'.
ω: Are CubeSat's angular rates.
I: Inertia Tensor.
Td: Disturbance Torque.
U: Torque input.

B. Perturbations and disturbances

The perturbations and disturbances are external and random torques that affect our plant. In this project, two types of disturbances are considered: Constant disturbances (4) and miscellaneous disturbances (5).

$$Td = [Td_x, Td_y, Td_z]^T \quad (4)$$

$$Td = A[\sin(f_x t), \sin(f_y t), \sin(f_z t)]T \quad (5)$$

C. Controller

The controller is an algorithm capable of reaching and keeping the desired attitude or set point [1]. In this project, two control algorithms are implemented and simulated to make a comparison between them. The two chosen algorithms are the Boskovic controller based on the variable structure approach [4] and the feedback controller explained in [3].

D. *Controller performance*

To make a fair comparison between control laws, [2] presents six performance parameters. However, in this project, four parameters are implemented and shown in the main GUIDE. The following parameters are implemented:

- **Settlement time at 5%.** Is the time elapsed from the application of an ideal instantaneous step input to the time at which the system output has entered and remained within a specified error band (5%).
- **Error Euler Angle Integration (EULERINT).** It is the integral of the error angle about the Euler axis of rotation. This shows the accumulated angle error.
- **Average of Square of the Commanded Torque (ASCCT).** It is a measure of magnitude equivalent to the effective average torque exerted on the three satellite axes
- **Computational Cost (O).** The average time that the algorithm takes to calculate the new control command.

III. Design and Implementation

The "CubeSat Attitude Control Evaluator" consists of five windows to give the users an easy interface with no data overcharging. The main window has six parts, and Figure 2 depicts that.

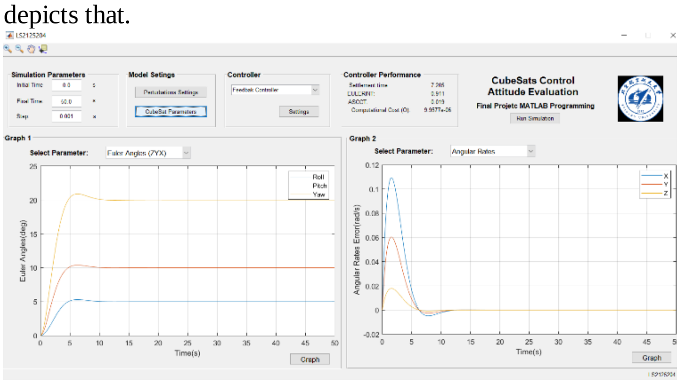

Figure 2:Window1: Main GUIDE window. (LS2125204.fig)

A. *Part 1: Simulation Parameters*

This part contains the parameters related to the system simulation like initial time, final time, and the integration step. Figure 3 depicts this part with the tested values.

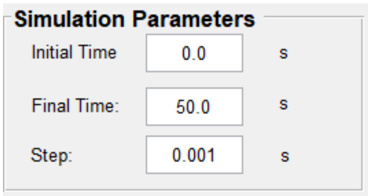

Figure 3: Main Guide: Simulation Parameters part.

B. *Part 2: Model settings*

This part contains two buttons that open other windows with the CubeSat parameters and the external perturbation settings. Figure 4 depicts the implemented two buttons and are described as follows:

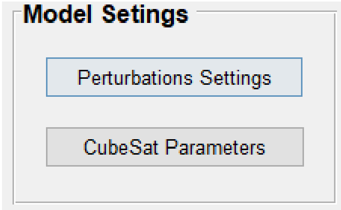

Figure 4: Model settings part.

i) Button1: Perturbation Settings

In this window, users can choose between two perturbations types: Constant perturbation and Miscellaneous perturbations. The first is a constant three-component torque vector, and the latter consists of a sinusoidal perturbation torque vector with three components. Figure 5 depicts this window with the default values.

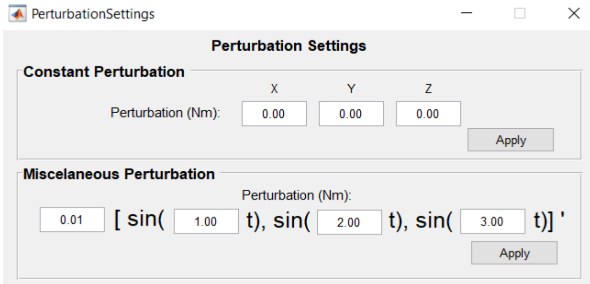

Figure 5: Window 2: Perturbation Settings.(PerturbationSettings.fig)

ii) CubeSat Parameters

In this window, we can configure the inherent properties for a CubeSat modeled as a rigid body. The parameters that users can set are the Inertia tensor and initial conditions of the system. Figure 6 depicts this window with the default values.

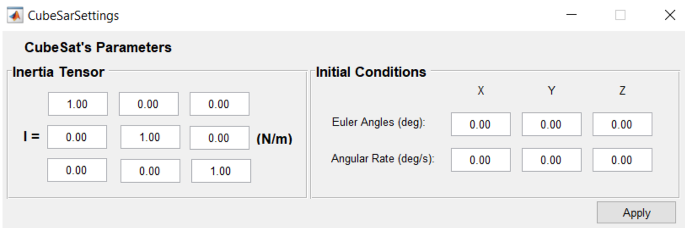

Figure 6: Window3: CubeSat's Parameters. (CubSarSettings.fig)

C. *Part3: Controller Settings*

This part contains a popup menu where users can select between two control algorithms. The button "settings" open another window in which users can customize controller configurations and gains. Figure 7 depicts this part with the default value.

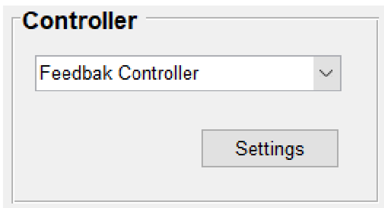

Figure 7: Controller settings part.

Figure 8 shows the Boskovic controller Settings window opened when the Boskovic Controller is selected.

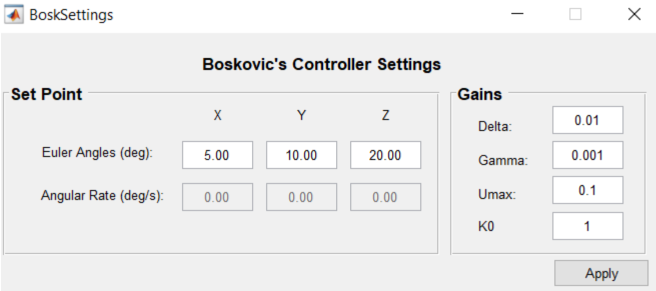

Figure 8: Window4: Boskovic's Controller Settings window with default values. (BoskSettings.fig)

Figure 9 shows the Feedback controller Settings window opened when the FeedBack Controlled is Selected.

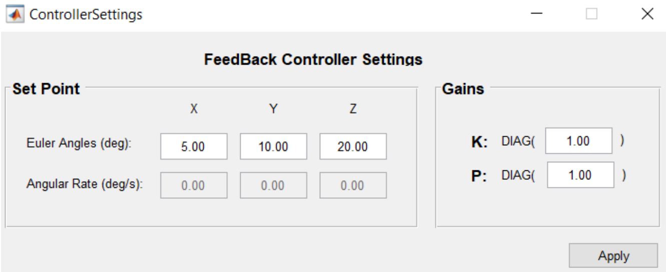

Figure 9: Window5: Feedback Controller Settings with default values. (ControllerSettings.fig)

D. *Part 4: Controller Performance*

This part displays the four parameters explained in section II. These parameters are helpful to evaluate the performance of each Control algorithm and are updated after a new simulation is accomplished. Figure 10 depicts this part with some values calculated with the default parameters.

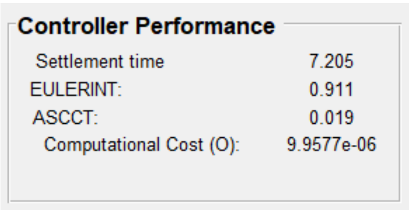

Figure 10: Controller performance parameters.

E. *Part 5: Graphic 1 and 2*

Graphics 1 and 2 display many time-variant parameters from simulations. The parameters available to be graphed are: 1) Euler Angles, 2) Quaternions, 3) Angular Rates, 4) Perturbations, 5) Control torque, 6) EULERINT, 7) ASCCT, and 8) K parameter. Figure 11 depicts Graphic 1 plotted with the default value.

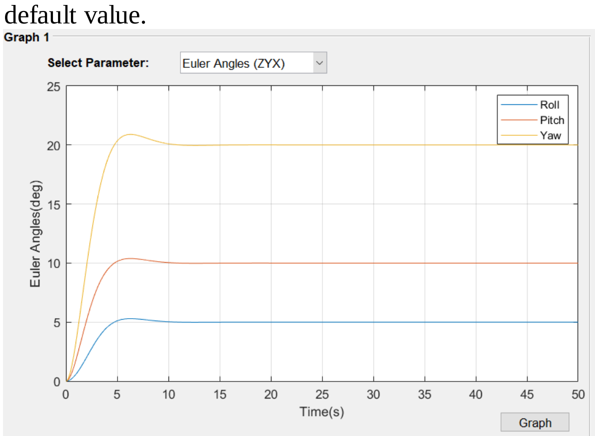

Figure 11: Graphic 1 with Euler Angles. Graphic 2 is similar.

F. *Part 6: Run Button*

The run button reads all the settled values from the parameters previously mentioned and starts to perform the simulation. Figure 12 depicts the simulation button.

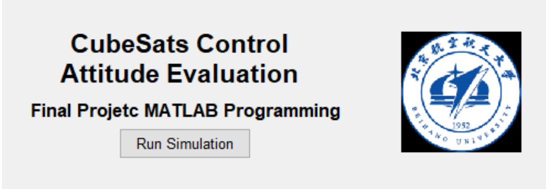

Figure 12: Main guide tittle and Run Button

IV. Testing

To test the present project, a simulation with the parameters of Table I is performed.

TABLE I. PROJECT PARAMETERS

Simulation Parameters	
Initial Time	*0.0*
Final Time	*50.0*
Step	*0.001*
CubeSat's Parameters	
Inertia Tensor	$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$
Initial Conditions	*Euler Angles (deg): [10.00,10.00,0.00]* *Angular Rate (deg/s): [0.00,10.00,0.00]*
Perturbation Settings	
Constant Perturbation (Nm)	*[0.00, 0.00, 0.00]*
Miscellaneous Perturbation (Nm)	*0.05[sin(1.00t), sin(2.00t), sin(3.00t)]*
Controller Settings	
Set Point (deg)	*[10.00, 50.00, 30.00]*
Boskovic's Controller Settings	*Delta = 0.01* *Gamma = 0.001* *Umáx = 0.1* *K0=1*
Feedback Controller	*K= eye(3)* *P=eye(3)*

As can be seen in Figure 13 every parameter in Table I is put into the GUIDE using the parts and buttons described in Section III. Finally, we test each simulation applying constant perturbations and Miscellaneous perturbations for both controllers.

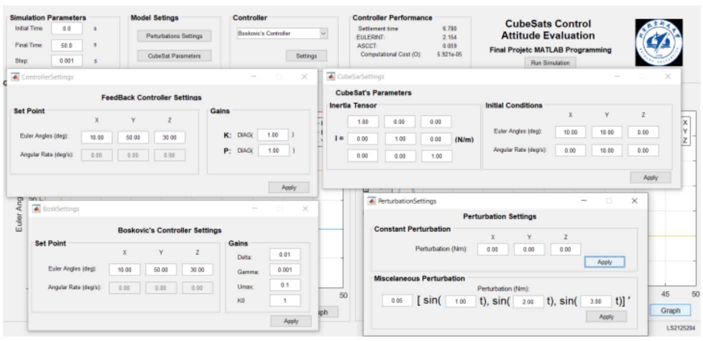

Figure 13: Parameters put into the GUIDE to perform the simulations.

A. *First test: Feed Back controller With constant perturbations*

To test the Feedback controller and apply constant perturbations, we choose the "Feedback Controller" in the popup menu and click on the “perturbation settings” button. Then, click on the “Apply” button located in the “Constant Perturbation” panel. Finally, click on the “Run simulation” button to begin simulation and obtain the results of Figures 14, 15, 16, and 17.

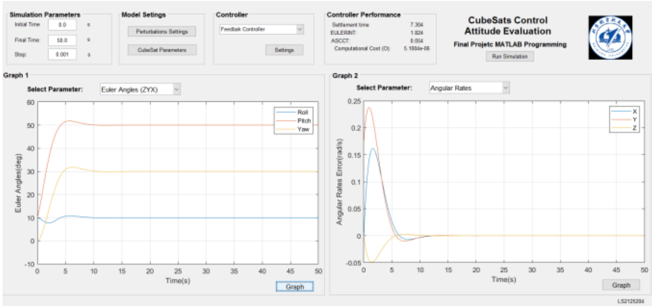

Figure 14: Euler Angles and Angular rates are values plotted by default.

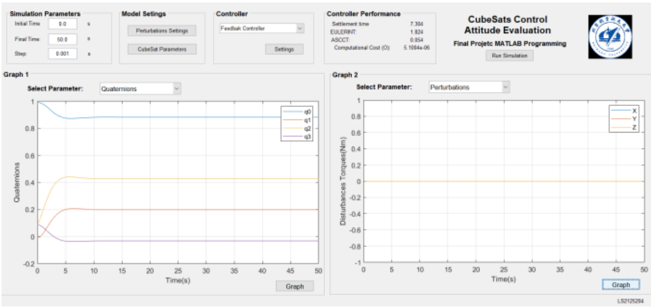

Figure 15: Quaternions and perturbations.

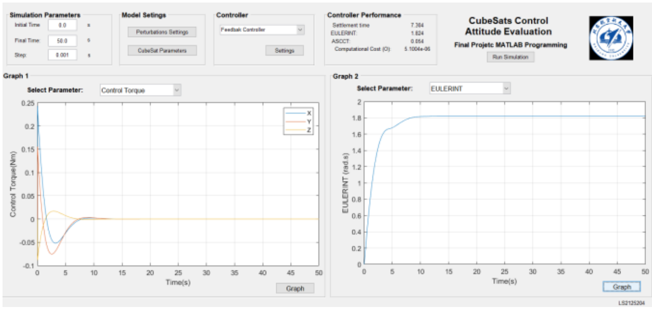

Figure 16: Control Torque and EULERINT.

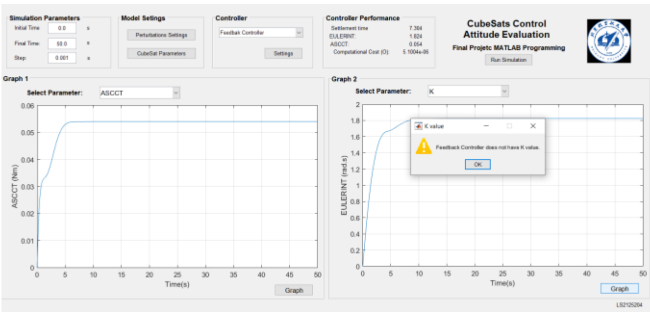

Figure 17: ASCCT and K value.

It is important to mention that to change the plotted parameter, it is necessary to select the desired parameter in the popup menu and then click on the “Graph” button.

B. Second Test: Boskovic's controller With constant perturbations

To test the Boskovic's controller and apply constant perturbations, we choose the “Boskovic's Controller” in the popup menu and click on the “perturbation settings” button. Then, click on the “Apply” button located in the “Constant Perturbation” panel. Finally, click on the “Run simulation” button to begin simulation and obtain the results of Figure 18.

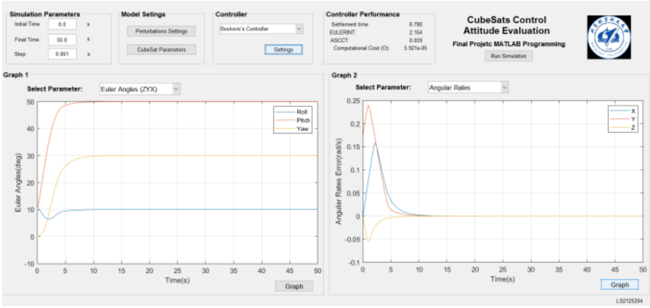

Figure 18: Euler Angles and Angular rates of Boskovic Controller.

C. Third test: Feed Back controller With miscellaneous perturbations

To test the Feedback controller and apply miscellaneous perturbations, we choose the “Feedback Controller” in the popup menu and click on the “perturbation settings” button. Then, click on the “Apply” button located in the “Miscellaneous Perturbation” panel. Finally, click on the “Run simulation” button to begin simulation and obtain the results of Figure 19.

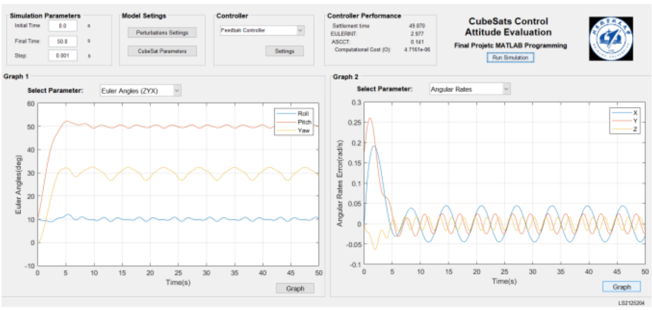

Figure 19: Feedback Controller under miscellaneous torques.

D. Fourth test: Boskovic's controller with miscellaneous perturbations

To test the Boskovic's controller and apply miscellaneous perturbations, we choose the “Boskovic's Controller” in the popup menu and click on the “perturbation settings” button. Then, click on the “Apply” button located in the “Miscellaneous Perturbation” panel. Finally, click on the “Run simulation” button to begin simulation and obtain the results of Figure 20.

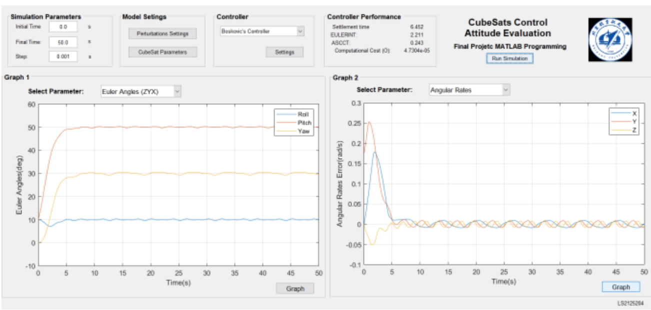

Figure 20: Boskovick's controller under miscellaneous torques.

Finally, we can notice that the K parameter is only available to be plotted when the Boskovic's controller is selected. The reason for this is because Feedback controller doesn't have a K parameter.

V. CONCLUSION

In this project, the GUIDE MATLAB tool solves the engineering problem of "attitude control simulation for small satellites". The presented GUIDE has an easy-to-use interface capable of modifying control gains and simulating two different control laws under constant and miscellaneous disturbances. Also, it is capable of measuring four performance parameters necessary to evaluate and compare the tuning and designed algorithms. Finally, we can confirm that engineers can perform several tests with many parameter combinations using this tool. However, other control algorithms could be added to this project to make a robust program.

REFERENCES

[1] F. L. Markley and J. L. Crassidis, Fundamentals of spacecraft attitudedetermination and control. New York: Springer, 2014.

[2] Brayan Espinoza Garcia, Ayrton Martin Yanyachi and Pablo Raul Yanyachi, “Non-Linear Control Strategies for Attitude Maneuvers in a CubeSat with Three Reaction Wheels” International Journal of Advanced Computer Science and Applications (IJACSA), 11(11), 2020.

[3] Schaub, Hanspeter et al., “Feedback control law for variable speedcontrol moment gyros,” Journal of the Astronautical Sciences, 1998,vol. 46, no 3, p. 307-328.

[4] J. D. Boskovic, S. Li, and R. K. Mehra, “Robust tracking control designfor spacecraft under control input saturation,” Journal of Guidance,Control, and Dynamics, 2004, vol. 27, no 4, p. 627-633