Design of Feedback Controllers for Control Moment Gyroscopes

Evan Lee*
University of Illinois Urbana Champaign, Champaign, IL, 61820

Spacecraft attitude control requires specific methods that take into account the fact that a spacecraft has nothing to physically push off of in space or in orbit. Control Moment Gyroscopes (CMGs) are one of many potential moment control systems that allow a spacecraft to control its attitude without pushing off of another physical object. These systems work by rotating a wheel within the spacecraft, then applying torque to that wheel, which has a high angular momentum. The wheel at the same time applies an equal and opposite torque to the spacecraft, which can thus be moved and control its attitude. This work explores the viability of the use of CMGs on real-world spacecraft through analysis of the behavior and potential controllability of systems in general, followed by analysis of a hypothetical CMG. The CMG was modelled by deriving (idealized) equations of motion through dynamics, then linearizing about a chosen equilibrium point. The resulting equations were then used to create another equation modeling our system in state space form. By placing eigenvalues in this state space using Ackermann's Method, the input, in this case torque applied to the wheel's gimbal, necessary to make the system asymptotically stable was calculated, and a controller was built that ensures the system stabilizes from many initial conditions, with some error expected due to our idealizations. The resulting controller was applied successfully close to equilibrium and less successfully farther from equilibrium. This work concludes that CMGs are well suited for making adjustments of $< \frac{\pi}{2}$ to spacecraft attitude about one axis, but larger changes in attitude and stabilization from larger initial perturbations are not possible with only CMGs.

I. Nomenclature

A	=	state matrix
B	=	input matrix
x	=	state vector
$\dot{x}$	=	rate of change of state vector
u	=	input vector
K	=	gain matrix such that $u = -Kx$
W	=	controllability matrix
M_{CCF}	=	a matrix in controllable canonical form
$\sigma(M)$	=	set of eigenvalues for matrix M
q_1	=	platform angle
$\dot{q}_1$	=	platform angular velocity
$\ddot{q}_1$	=	platform angular acceleration
q_2	=	gimbal angle
$\dot{q}_2$	=	gimbal angular velocity
$\ddot{q}_1$	=	platform angular acceleration
f	=	vector equal to $\dot{x}$ equations
J_{ij}	=	moment of inertia
m	=	mass
r	=	radius
g	=	gravity
v_{rotor}	=	angular velocity of rotor

*Undergraduate Student, Department of Aerospace Engineering, 104 S. Wright St, Urbana, IL 61801

II. Introduction

SPACECRAFT attitude control is made difficult by the fact that spacecraft are ideally not in physical contact with anything, even air, and thus have nothing to push off of to change their motion or attitude. Systems that allow spacecraft to control their attitude given their unique situation include propulsion, reaction wheels, and the focus of this work: the Control Moment Gyroscope (CMG). CMGs work by rotating a wheel attached through a gimbal to the spacecraft and then applying torque to said wheel. Since the wheel has a high angular momentum, it can be subjected to large amounts of torque without much change in this angular momentum. The wheel thus applies an equal and opposite reaction torque to the spacecraft, which allows it to move itself, changing its attitude without pushing off anything externally.

In this work, the predictability and controllability of CMGs are explored in the context of these systems' viability for use on spacecraft. This analysis is conducted through linearizing the dynamical equations of a hypothetical CMG about a set of equilibrium points to create a model of the system in state space form. Ackermann's Method for placing eigenvalues was then used to find a gain matrix that would ensure stability and give the necessary input in terms of the state at a point in time. Finally, this gain matrix was used to build a controller that could calculating the necessary input from the current state. The resulting controller was tested on a variety of inputs both close to and far from the chosen equilibrium point, with success close to the equilibrium point and failure far from it, and conclusions were drawn from these results.

In Section III, the process for creating a linearized model and building a controller from this model is outlined in detail from start to finish. In Section IV, the application of this process to a hypothetical CMG is conducted. In Section V, the resulting behavior of the model from Section IV, along with its successes and failures, is reported. It is shown that CMGs are well suited for attitude changes $< \frac{\pi}{2}$, but CMGs are unable to stabilize larger changes and initial perturbations from equilibrium.

III. Model

In order to assess the viability of CMGs for use on spacecraft, their behavior must first be modeled so it can be predicted, and so a controller can be built to determine a certain CMG's behavior. Only after achieving this and evaluating the results can a judgement be made.

A. Linearization

There are many ways to model a dynamical system, but this work uses only the method of representing it in state space form. In this form, the rate of change of the state vector is equal to the state matrix A multiplied by the state vector x added to the input matrix B multplied by the state vector u.

$$\dot{x} = Ax + Bu \tag{1}$$

Most dynamical systems do not naturally conform to this form, which represents the dynamics in a signle linear, first-order, ordinary differential equation, so the system must be linearized in order for it to be representable in state space form. This is done by finding equations for the highest-order derivatives from the dynamical equations of a system. A state vector x is then designed such that it contains every state besides the highest order derivative states, and the rate of change state vector $\dot{x}$ has these highest order derivatives as its own highest order elements. For example, if the equations for $\ddot{q}_1$ and $\ddot{q}_2$ are known, as they are for the CMG model used later, the x and $\dot{x}$ are as follows.

$$x = \begin{vmatrix} q_1 \\ \dot{q}_1 \\ q_2 \\ \dot{q}_2 \end{vmatrix} \quad \text{and} \quad \dot{x} = \begin{vmatrix} \dot{q}_1 \\ \ddot{q}_1 \\ \dot{q}_2 \\ \ddot{q}_2 \end{vmatrix} \tag{2}$$

Note that the first and third elements of the rate of change of state vector are the second and fourth elements of the state vectors, i.e. they are state variables. The input vector u is just a vector of inputs we can control in the system; these inputs may or may not be in the equations for the highest order state derivatives.

Once x, $\dot{x}$, and u are found, the the state and input matrices must be found as well. This is done by first assembling a vector f of the elements of $\dot{x}$, where the highest order elements are replaced by their equations in terms of the state variables, which we found from the dynamical equations. We then choose an equilibrium point about which the system will be linearized. This point is a set of values of the state variables such that when f is evaluated at equilibrium, $f = \vec{0}$. x is then set equal to the state variables minus their respective equilibrium values. u is similarly adjusted: the inputs are the input variable minus the inputs at equilibrium. The state matrix A is equal to the Jacobian of f w.r.t. the state variables, evaluated at equilibrium (this is why we represented the rate of change of state vector as a function of the state variables). The input matrix B is equal to the Jacobian of f w.r.t. the input variables, also evaluated at equilibrium. With numerical values for A, B, x, $\dot{x}$, and u, our system is linearized and in state space form.

B. Asymptotic Stabilization and Control

While linearization creates a model that can predict the behavior of a dynamical system, it doesn't provide a way to control this behavior. One method of controlling a linearized dynamical system is to choose an equilibrium point, then change the linearized system such that the state always converges to $\vec{0}$, which is synonymous with converging to equilibrium since we have linearized about an equilibrium point. Thes behavior is caused asymptotic stability, and it occurs naturally in some systems, and must be imposed on others.

Before attempting to change a linearized system in this way, some properties of the matrix exponential must be discussed. In general, if $\dot{x} = Mx$, then $x(t) = e^{Mt}x(0)$. Due to matrix exponential properties, if the matrix M can be diagonalized such that $M = VSV^{-1}$, then $e^{Mt} = Ve^{St}V^{-1}$ and most importantly

$$e^{St} = \begin{vmatrix} e^{s_1 t} & 0 & \cdots & 0 \\ & e^{s_2 t} & 0 & \vdots \\ & & \ddots & 0 \\ sym. & & & e^{s_n t} \end{vmatrix} \tag{3}$$

In combination with the fact that a matrix can be represented as its eigenvalues in diagonal matrix S, and a matrix of eigenvectors V, the upshot of all this is that if $\sigma(M) \in \mathbb{C}^-$, $\lim_{t\to\infty} e^{Mt}x(0) = 0$.

Returning to CMGs, if $\dot{x} = Ax + Bu$ could instead be represented as $\dot{x} = Mx$, and $\sigma(M) \in \mathbb{C}^-$, then $\lim_{t\to\infty} x(t) = 0$. In other words, the linearized system would be asymptotically stable. To impose this on the state space form equation, a gain vector K must be found such that $u = -Kx$, thus making $\dot{x} = (A - BK)x$. If K is such that $\sigma(A - BK) \in \mathbb{C}^-$, then the system is made asymptotically stable.

To find K such that $\sigma(A - BK)$ is the set of desired eigenvalues Σ, Ackermann's Method is used. First, define the desired characteristic polynomial using the values of Σ.

$$(\lambda - \Sigma_1) \cdots (\lambda - \Sigma_n) = \lambda^n + \alpha_1 \lambda^{n-1} + \cdots + \alpha_n \tag{4}$$

Then, find the characteristic polynomial of A.

$$\sigma(\lambda I - A) = \lambda^n + \beta_1 \lambda^{n-1} + \cdots + \beta_n \tag{5}$$

Next, define the controllability matrix W. If W is not invertible, eigenvalues cannot be placed anywhere.

$$W = \begin{bmatrix} B & AB & \cdots & A^{n-1}B \end{bmatrix} \tag{6}$$

Then, find A, B, and W in controllable canonical form.

$$A_{ccf} = \begin{vmatrix} -\beta_1 & -\beta_2 & \cdots & -\beta_{n-1} & -\beta_n \\ 1 & 0 & \cdots & 0 & 0 \\ 0 & 1 & \cdots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & 1 & 0 \end{vmatrix}, \quad B_{ccf} = \begin{vmatrix} 1 \\ 0 \\ 0 \\ \vdots \\ 0 \end{vmatrix} \quad \text{and} \quad W_{ccf} = \begin{bmatrix} B_{ccf} & A_{ccf}B_{ccf} & \cdots & A_{ccf}^{n-1}B_{ccf} \end{bmatrix} \tag{7}$$

Next, define L as follows using α and β values from our characteristic polynomials.

$$L = \begin{bmatrix} \alpha_1 - \beta_1 & \alpha_2 - \beta_2 & \cdots & \alpha_n - \beta_n \end{bmatrix} \tag{8}$$

Finally, use W_{ccf} and W to get K from L.

$$K = LW_{ccf}W^{-1} \tag{9}$$

Mathematically, this means that K can be found from A, B, and Σ such that when $u = Kx$, $\sigma(A - BK) = \Sigma$. Thus, it is possible to place eigenvalues Σ such that $\sigma(A - BK) \in \mathbb{C}^-$, and $\lim_{t\to\infty} x(t) = 0$. *Physically*, this means that with K, it is possible at any point in time to calculate the input necessary to impose asymptotic stability from the state at that point in time, because of the relationship $u = -Kx$. A controller can thus be built using the gain vector K to calculate the necessary input to stabilize a system towards equilibrium.

IV. Design

In this work, the above method of linearization and stabilization was applied to create a model and controller for a CMG. First, x, $\dot{x}$, and f were found from the dynamical equations. Then, and equilibrium point was chosen and A and B were found from f. Lastly, K was found such that the system would be theoretically asymptotically stable. A controller was then built from K and tested using a simulation.

A. Finding State Space Matrices and Vectors from Dynamical Equations of a CMG

The CMG system used for this work consisted of a platform, with mass in a boom that extended from the rectangular platform, a gimbal, and a rotor housed and rotating inside said gimbal. It was assumed for simplicity that the velocity of the rotor would be constant and that the torque applied by the platform to the gimbal would be the only torque that needed to be accounted for. The parameters in the equations of motion are as follows:

- $m = 1.0$ kg, mass of boom
- $r = 2.0$ m, length of boom
- $g = 9.81$ m/s, acceleration of gravity
- $v_{\text{rotor}} = 500$ rad/s
- $J_{1z} = 0.5\ \text{kg}\cdot\text{m}^2$, principal moment of inertia of platform
- $J_{2x} = J_{2z} = 0.001\ \text{kg}\cdot\text{m}^2$, principal moments of inertia of gimbal
- $J_{3x} = J_{3y} = J_{3z} = 0.01\ \text{kg}\cdot\text{m}^2$, principal moments of inertia of rotor

In order to simplify the final dynamical equations, we define the following constants.

- $a_1 = -J_{3y} + 2J_{3z}$
- $a_2 = 2J_{3y}$
- $a_3 = -2gmr$
- $a_4 = 2J_{1z} + 2J_{2z} + 2mr^2$
- $a_5 = 2J_{3z}$
- $a_6 = \frac{J_{3y} - J_{3z}}{2(J_{2x} + J_{3x})}$
- $a_7 = -\frac{J_{3y}}{J_{2x} + J_{3x}}$
- $a_8 = \frac{1}{J_{2x} + J_{3x}}$

The dynamical equations that describe the motion of this CMG are as follows:

$$\ddot{q}_1 = \frac{a_1 \sin(2q_2)\dot{q}_1\dot{q}_2 + a_2\cos(q_2)\dot{q}_2 v_{\text{rotor}} + a_3\sin(q_1)}{a_4 + a_5\cos^2(q_2)} \quad [1] \tag{10}$$

$$\ddot{q}_2 = a_6\sin(2q_2)\dot{q}_1^2 + a_7\cos(q_2)\dot{q}_1 v_{\text{rotor}} + a_8\tau \quad [1] \tag{11}$$

Using these equations, x and $\dot{x}$ were calculated to reflect these dynamical equations.

$$x = \begin{vmatrix} q_1 \\ \dot{q}_1 \\ q_2 \\ \dot{q}_2 \end{vmatrix} \quad \text{and} \quad \dot{x} = \begin{vmatrix} \dot{q}_1 \\ \ddot{q}_1 \\ \dot{q}_2 \\ \ddot{q}_2 \end{vmatrix} \tag{12}$$

f was then calculated by representing $\dot{x}$ as a function of the state variables.

$$f = \begin{vmatrix} \dot{q}_1 \\ \frac{a_1 \sin(2q_2)\dot{q}_1\dot{q}_2 + a_2 \cos(q_2)\dot{q}_2 v_{\text{rotor}} + a_3 \sin(q_1)}{a_4 + a_5 \cos^2(q_2)} \\ \dot{q}_2 \\ a_6 \sin(2q_2)\dot{q}_1^2 + a_7 \cos(q_2)\dot{q}_1 v_{\text{rotor}} + a_8 \tau \end{vmatrix} \tag{13}$$

With f calculated, an equilibrium point could be chosen. Originally, attempts were made to choose the equilibrium point strategically. $\dot{q}_1$ and $\dot{q}_2$ has to equal 0 as they were the only components of f_1 and f_3 respectively. τ also had to equal 0 as it was only multiplied by a constant in f_4. However, the first attempted equilibrium state variables were chosen so the not only $\dot{q}_1$, $\dot{q}_2$, and τ were 0, but q_1 and q_2 were chosen such that every trigonometric function in f would evaluate to 0 as well. This was going too far, as the partial derivatives in the Jacobian of f w.r.t. the state variables would also equal 0 at equilibrium, leading to A having too many zeros and fouling up inversions required by Ackermann's Method. Eventually, a series of random guesses led to the following equilibrium values, which allowed Ackermann's Method to proceed.

- $q_{1e} = 0$
- $\dot{q}_{1e} = 0$
- $q_{2e} = 0$
- $\dot{q}_{2e} = 0$
- $\tau_e = 0$

After the fact, it was realized that these values succeeded because not all of the trig functions evaluated to 0, so after they were isolated from states that had to be 0 after the Jacobian was calculated, they would not be equal to 0 themselves, allowing A to have more non-zero values.

With f and the equilibrium values calculated, A and B could be calculated using the method outlined in Section II. The partial derivatives for the Jacobian and values evaluated at equilibrium were calculated by implementing the method in Python.

$$A = \begin{vmatrix} 0 & 1 & 0 & 0 \\ -4.349 & 0 & 0 & 1.108 \\ 0 & 0 & 0 & 1 \\ 0 & -454.545 & 0 & 0 \end{vmatrix} \quad \text{and} \quad B = \begin{vmatrix} 0 \\ 0 \\ 0 \\ 90.91 \end{vmatrix} \tag{14}$$

B. Designing a Controller for a CMG

With A and B calculated, K could be calculated using Ackermann's Method, once again implemented in Python. With the following set of desired eigenvalues Σ, the following K was calculated.

$$\Sigma = \begin{vmatrix} -1 & -2 & -3 & -4 \end{vmatrix} \quad \text{results in} \quad K = \begin{vmatrix} 0.0646 & -4.75 & 0.0607 & 0.11 \end{vmatrix} \tag{15}$$

Calculating the eigenvalues of $A - BK$, the results show that $\sigma(A - BK)$ is indeed $(-1, -2 - 3, -4) \in \mathbb{C}^-$. The closed loop controlled system should theoretically be asymptotically stable.

K can be implemented in a controller in the simulation provided by the instructors. In the "run" function, the state is passed into the function and an instruction for torque applied to the gimbal within the range $[-1, 1]$ is expected. Since this gimbal torque is the only input for the system, we can implement a controller here by setting the desired gimbal torque $u = -Kx$. This will ensure that the gimbal torque is at least instructed to be equal to the torque necessary to stabilize the system according to the calculated model. In other words, $\dot{x} = (A - BK)x$, and the system should stabilize to our equilibrium point. With the controller build, a series of tests were conducted to confirm or deny the controller's ability to stabilize the CMG.

V. Results

Tests for the controller were conducted using simulator provided by the instructors. To test how well the controller can stabilize the CMG system, the system must start with a perturbation from which the controller must recover successfully. The four tests in this work varied the initial platform angle, perturbing the other state variables to challenge the controller, but keeping these consistent across the tests. Specifically, $\dot{q}_1 = 0.5$ rad/s, $q_2 = \frac{\pi}{2}$ rad, and $\dot{q}_2 = 0.3$

rad/s across all the conducted tests. q_1 was tested at $\frac{\pi}{6}$, $\frac{\pi}{4}$, $\frac{\pi}{3}$, and $\frac{\pi}{2}$. The platform angle was chosen to be varied because changing the angle of the platform would be the ultimate goal of a CMG in spaceflight. These tests would also offer insight into how the initial conditions of the CMG system affected its motion. All four simulations ran for 30 seconds each for consistency. All state variables, the gimbal torque command, actual gimbal torque applied, and the rotor velocity were all recorded over the test.

For the first two initial platform angles tested, $\frac{\pi}{6}$ and $\frac{\pi}{4}$, the controller performed successfully. The platform oscillated at first, but the magnitude of the oscillations in both tests decreased over time until the platform eventually came to rest at the chosen equilibrium point, with the $q_1 = \frac{\pi}{4}$ test taking slightly longer to stabilize. This was likely because the perturbation was larger for this test. Regardless, the system was successfully stabilized in both cases.

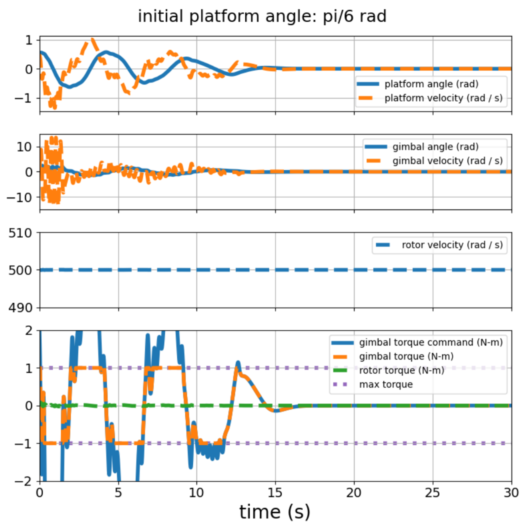

Fig. 1 Controller test when $q_1 = \frac{\pi}{6}$

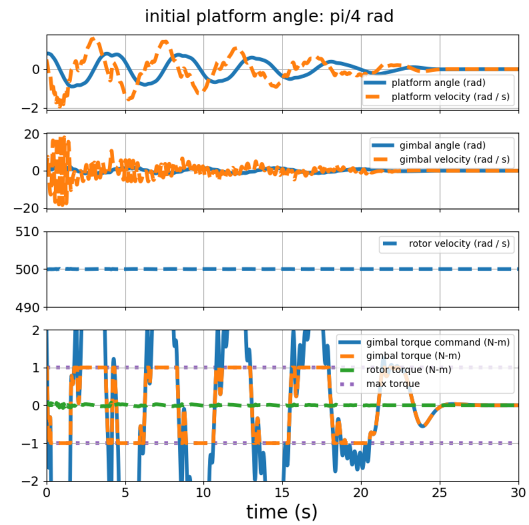

Fig. 2 Controller test when $q_1 = \frac{\pi}{4}$

For the larger perturbations of $\frac{\pi}{3}$ and $\frac{\pi}{2}$, the controller was not successful. For the test with initial platform angle of $\frac{\pi}{3}$, the oscillation of the platform did not decease and in fact slightly increased; the controller was completely ineffective. For the test with initial platform angle of $\frac{\pi}{2}$, the controller, and indeed the simulation itself, broke completely. The platform spun rapidly and unpredictably.

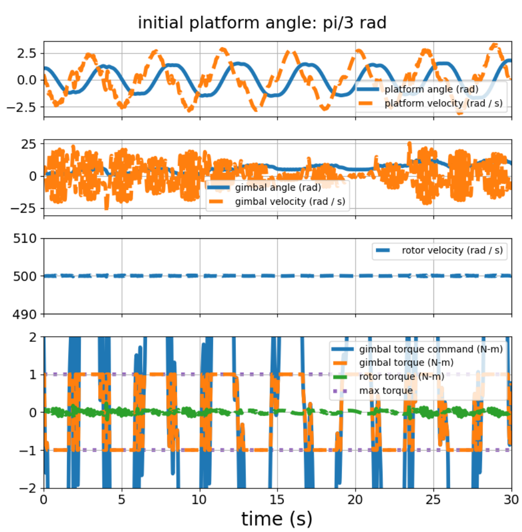

Fig. 3 Controller test when $q_1 = \frac{\pi}{3}$

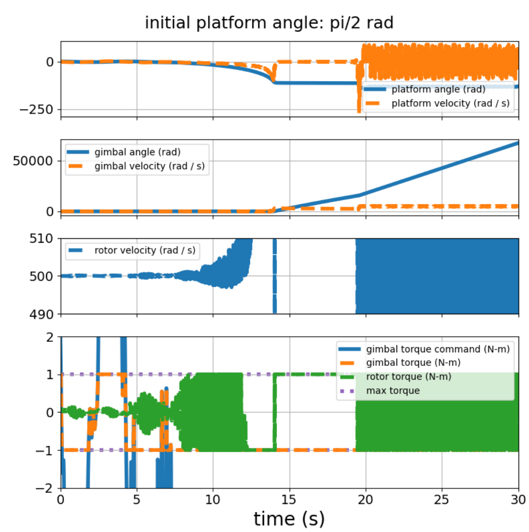

Fig. 4 Controller test when $q_1 = \frac{\pi}{2}$

There are many possible explanations for this behavior. The first and most obvious is that the farther away from the equilibrium point the system gets, the less accurate the calculated model, which was linearized about this point, will be. A more specific possible explanation is that the gimbal torque required by the controller to stabilize the system after these initial perturbations was much higher in magnitude than the limits set by the simulation. If this was the case, the controller would never have the power to begin stabilizing the system and would be powerless to stop it going out of control, potentially even contributing to the chaos.

Regardless of the specific causes, it is clear that the controller can stabilize the system when small to medium sized perturbations from equilibrium are the initial conditions, but it does not work past a certain distance from the chosen equilibrium. Whether the initial conditions are close ($< \frac{\pi}{4}$) or far from equilibrium is key to the performance of the controller. The initial conditions also affect the motion itself, with initial conditions close to equilibrium resulting in stabilization and initial conditions far from equilibrium resulting in oscillation or completely erratic motion.

References

[1] Ornik, M., and Puthumanaillam, G., "Design Project 1," , 2023. URL `https://github.com/uiuc-ae353/ae353-fa23/wiki/Design-Project-1`.