ME 4012 Project Report: Cue-B

Will Compton, Domenic DiCarlo, Thalmus McDowell, and Micah Morris
ME 4012 Modeling and Control of Motion Systems

Abstract

Taking inspiration from the project Cubli [1], Team A2 sought to design and construct a cube capable of balancing on an edge using a reaction wheel. Balance through the wheel was achieved using state space and pole placement via Ackerman's method. The control system was implemented through an Arduino Teensy and ODrive motor and motor controller. Ultimately, balance was achieved along with adequate disturbance rejection. Future improvements could include a control scheme to make the cube stand up onto an edge from an adjacent face, saturation avoidance through the inclusion of wheel velocity as a state, and potentially additional reaction wheels to achieve vertex balance alongside edge balance.

Introduction

Team A2 has decided to develop a reaction wheel-controlled cube that launches from rest to balancing on an edge. Balancing items is inherently unstable, as slight adjustments to position can cause tipping and falling; therefore, to balance without external input, control mechanisms will need to be put into place. This project was inspired by demonstrations of the research project Cubli, a small self-balancing cube that is capable of not only balancing on edges but also balancing on vertices as well as "walking" by rolling itself from face to edge to adjacent face. The dexterity of Cubli was impressive, and the team sought to emulate it in more conservative respects.

Design

The components of Cue-B that were purchased are the motor and motor controller. The Arduino Teensy 4.1, IMU, Encoder, battery, and ball bearings were all components that the group already owned. The parts that were manufactured include the wheel, the frame of Cue-B, and the motor mount. The main components of Cue-B are the reaction wheel, motor, and IMU. The torque from the reaction wheel is used to balance and control Cue-B. The shape of the reaction wheel was optimized to maximize the moment of inertia with the minimal amount of mass possible. Bolts were distributed along the outer ring of the wheel to increase its inertia. The motor drives the reaction wheel to stabilize Cue-B. It is controlled by an O-drive. The O-drive is great because it has a torque mode that allows direct control of the motor's torque. The IMU determines the orientation of the cube so that the reaction wheel can respond accordingly. The IMU was placed on the pivot point of Cue-B to get the most accurate readings. Another important design choice is the location of the battery; it was placed on the pivot point as well. The battery's location makes it easier for Cue-B to be controlled because it lowers the center of mass. Therefore, less torque will be needed to correct Cue-B when it falls from the equilibrium position. The setup of Cue-B can be found in Figure 1.

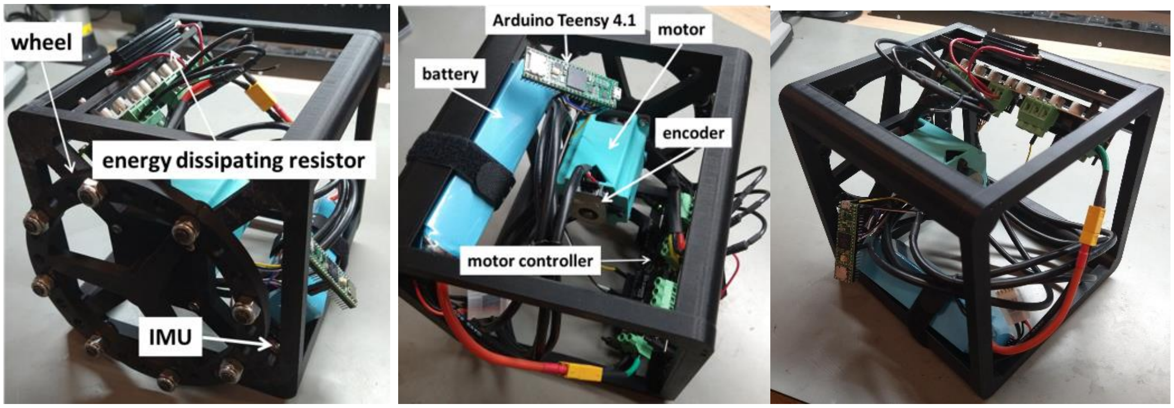


Figure 1: *Mechanical design and Electrical components of the Cue-B.*

Modeling

Cue-B was modeled as a square with sharp corners. To determine the dynamic equations a free body diagram was created, illustrated in Figure 2:

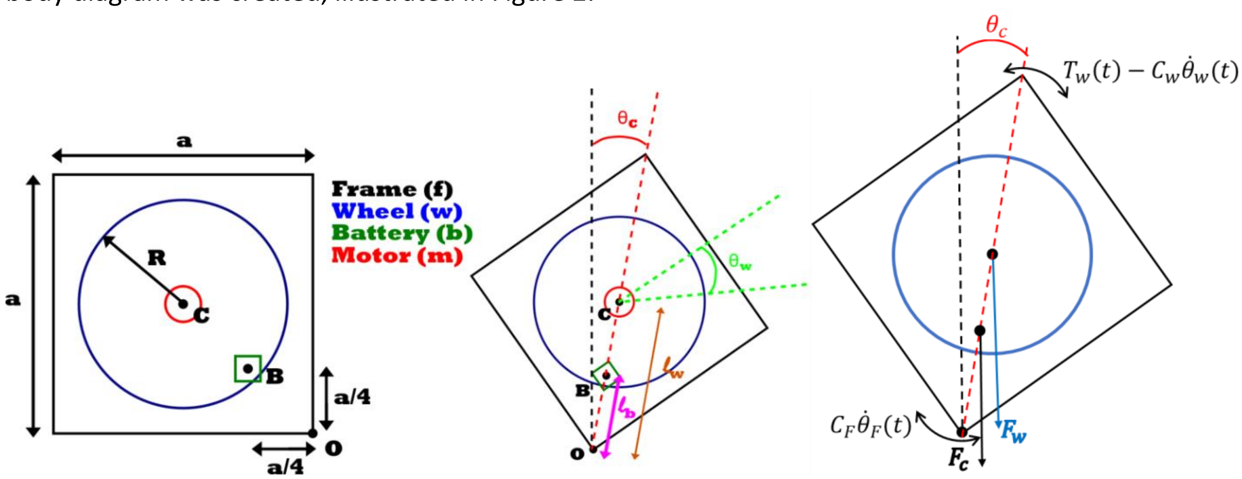


Figure 2: *Cue-B parameters and free body diagram*

As shown above, the free body diagram uses a square rotated θ_c from the origin. This represents the instantaneous moment when the cube is rotated away from its balance point. This orientation was chosen because Cue-B is meant to be controlled about its balancing point, which is represented by $\theta_c = 0$. Two moment balance equations were used to determine the dynamic equations. Equation 1 was created from a moment balance about point O. Equation 2 was created from the moment on the wheel. The dynamic equations for the system in Figure 2 are,

$$\sum M_c^o = I_O^c \ddot{\theta}_c(t) = m_w g l_w \sin(\theta_c) + m_c g l_c \sin(\theta_c) - T_w(t) + C_w \dot{\theta}_w(t) - C_c \dot{\theta}_c(t) \qquad (1)$$

$$M_w = I_w \left(\ddot{\theta}_F(t) + \ddot{\theta}_w(t) \right) = T_w(t) - C_w \dot{\theta}_w(t) \qquad (2)$$

[2] was referenced to create the dynamic equations of motion. The variable definitions can be found in Table 1:

***Table 1**: Variable definitions*

Definition	Variable
Torque of the Wheel	T_w
Angle of Cube relative to the vertical	θ_c
Angle of wheel	θ_w
Mass of the frame	m_F
Mass of the battery	m_b
Mass of the motor	m_m
Mass of the wheel	m_w
Mass of the cube	$m_c = m_F + m_b + m_m + m_w$
Length to center of wheel	l_w
Length to the center of mass of the cube	l_c
Radius of the wheel	R_w
Radius of the motor	R_m
Length of the cube	a
Coefficient of friction on wheel axle	C_w
Coefficient of friction on rotation point of Cue-B	C_c
Acceleration due to gravity	g
Moment of Inertia of the Cube	$I_O^c = I_O^F + I_O^w + I_O^m + I_O^b$
Moment of Inertia of the Wheel	$I_O^w = m_w\left(R_w^2 + \frac{1}{2}a^2\right)$
Moment of Inertia of the Battery	$I_O^b = \frac{7}{24}m_b a^2$
Moment of Inertia of the Motor	$I_O^m = \frac{m_m}{2}(R_m^2 + a^2)$
Moment of Inertia of the Frame	$I_O^F = \frac{m_F}{6}\left(4a^2 + \frac{a^2}{4}\right)$

The length and radius measurements a, l_w, R_w and R_m was found using SolidWorks and a ruler. The length to the center of mass of the cube, l_c, was determined algebraically using the center of mass equation. The masses m_b and m_m was determined from the battery and motor provided specifications. The mass of the wheel and the frame, m_w *and* m_F, were weighed using a scale. The moments of inertia for the components, I_O^w, I_O^b, I_O^m, I_O^F, were determined using the known equations of similar geometries and then translating them from their center of mass to the rotation point O. For example, I_O^w was estimated using the moment of inertia of a hollowed cylinder translated to the pivot point O. The value of the coefficients of friction, C_w and C_c, was of the magnitude of 10^{-4}[2]. For simplification, the coefficients will be ignored when developing the state space representation.

The dynamic equations (1) and (2) involve a nonlinear term $\sin(\theta_c)$ thus, Equation 1 must be linearized. The equation was linearized about the balancing point $\theta_{c0} = 0°$ because that is the angle that Cue-B was controlled about. The resulting equation is,

$$I_O^c\ddot{\theta}_c(t) = m_w g l_w \theta_c + m_c g l_c \theta_c - T_w(t) + C_w\dot{\theta}_w(t) - C_c\dot{\theta}_c(t) \quad (3)$$

With the linearized moment equation of the cube, (3), and the moment of the wheel equation, (2), a Laplace transform was conducted to find the transfer function of the system. The Laplace transform was conducted in terms of $\theta_F(s), \dot{\theta}_w(s), and\ T_w(s)$. The desired transfer function is of the form $\frac{\theta_F(s)}{T_w(s)}$. This transfer function uses a torque input to control the desired angle of the cube. The motor used on Cue-B is torque controllable, which simplifies the transfer function because there is no need for a transfer function relating the angular velocity of the motor to the torque of the motor. The resulting transfer function is,

$$\frac{\theta_F(s)}{T_w(s)} = -\frac{s}{(C_w I_w + C_c)s^2 + (I_O^c - m_w g l_w - m_c g l_c + C_w C_c)s + C_w I_O^c - C_w m_w g l_w - C_w m_c g l_c} \tag{4}$$

From the dynamic equations a state space representation was also derived. Cue-B was controlled using state space techniques which will be expounded upon in the Controller Design section. The state space equations, given in Equations 5 and 6, simplify the model of the system by neglecting mechanical damping in the system.

$$\begin{bmatrix}\dot{\theta}_c\\ \ddot{\theta}_c\end{bmatrix} = \begin{bmatrix}0 & 1\\ \left(\frac{1}{I_{oc}}\right)(m_w l_w + m_c l_c)g & 0\end{bmatrix}\begin{bmatrix}\theta_c\\ \dot{\theta}_c\end{bmatrix} + \begin{bmatrix}0\\ \frac{1}{I_{oc}}\end{bmatrix}T_w = \begin{bmatrix}0 & 1\\ a & 0\end{bmatrix}\begin{bmatrix}\theta_c\\ \dot{\theta}_c\end{bmatrix} + \begin{bmatrix}0\\ b\end{bmatrix}T_w \tag{5}$$

$$Y = \begin{bmatrix}1 & 0\end{bmatrix}\begin{bmatrix}\theta_c\\ \dot{\theta}_c\end{bmatrix} \tag{6}$$

Controller Design

To design a state space controller, the entire state information must be accessible. To do this, partial observer is designed, the complementary filter, to combine acceleration and angular velocity measurements from the IMU into an estimate of the cube's orientation. Two methods exist to estimate orientation from the IMU: integration of angular velocity, shown in equation 7, and the geometry of the gravity vector, shown in equation 8, where both equations reference the quantities shown in Figure 3.

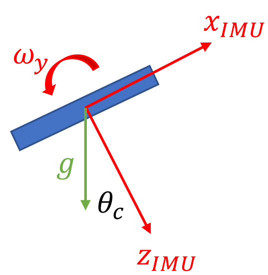


Figure 3: *The quantities measured by the IMU. Z and X reference the axes along which accelerations will be measured, and ω_y is the angular velocity about the remaining axis.*

$$\theta_{c,gyro}^i = \theta_c^{i-1} + \omega \Delta t \tag{7}$$

$$\theta_{c,accel}^i = tan^{-1}\frac{a_x}{a_z} \tag{8}$$

Because gyroscope measurements are prone to drift, the gyroscope method is accurate in the short term, and tracks high frequency movements well; however, its long-term accuracy and low-frequency characteristics are poor. Conversely, the acceleration method tracks low-frequency movements well, and has good long-term accuracy. The complementary filter, equation 9, combines both estimates in an infinite-impulse response filter [3].

$$\theta_c^i = \alpha\theta_{c,gyro}^i + (1-\alpha)\theta_{c,accel}^i, \quad 0 \ll \alpha < 1 \tag{9}$$

With the state estimated (as $\dot{\theta}_c = \omega_y$), the controller can be designed via pole placement. First, controllability must be checked:

$$C_M = [B \quad AB] = \begin{bmatrix} 0 & b \\ b & 0 \end{bmatrix} \rightarrow Full\ Rank, Controllable$$

Time specifications are given by desired system characteristics, at settling time of $t_s = 1$ and a maximum overshoot of $M_p = 0.05 = 5\%$:

$$M_p = 0.05 \rightarrow \zeta = \frac{\ln(M_p)}{\sqrt{\pi^2 + \ln^2 M_p}} = 0.69$$

$$t_s = 1 \rightarrow \omega_n = \frac{4}{t_s \zeta} = 5.79$$

Which gives rise to the following desired closed loop characteristic equation:

$$s^2 + 2\zeta\omega_n s + \omega_n^2 = s^2 + 8s + 33.5960 = 0$$

The poles of the closed loop system can then be placed accordingly.

$$\det(sI - (A - BK)) = \left| \begin{bmatrix} s & 0 \\ 0 & s \end{bmatrix} - \left(\begin{bmatrix} 0 & 1 \\ a & 0 \end{bmatrix} - \begin{bmatrix} 0 \\ b \end{bmatrix} [k_1 \quad k_2] \right) \right| = 0$$

$$\begin{vmatrix} s & -1 \\ -a + k_1 b & s + k_2 b \end{vmatrix} = s^2 + k_2 bs + a - k_1 b = 0$$

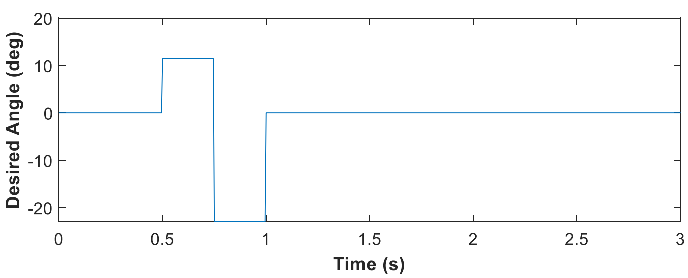
$$s^2 + k_2 bs + a - k_1 b = s^2 + 8s + 35.596 = 0$$

$$k_1 = 2.29 \quad k_2 = 0.201$$

A few details of the controller implementation deserve notes. First, due to wires, and asymmetry, the center of mass of the device is slightly misaligned with the z-axis of the IMU. To correct for this, the device is balanced by hand, and this offset angle is measured. The offset can then be subtracted so that the device balances over its center of mass instead of the IMU Z axis. Secondly, the control loop is implemented at a rate of 200Hz. It is assumed that this is a sufficiently fast update rate to ignore the discrete control aspects of the controller and treat it as continuous.

Experimental Results (and Simulation Results)

The closed loop system is first simulated to ensure that the designed response characteristics are correct. The results of the simulation are shown in Figure 4.



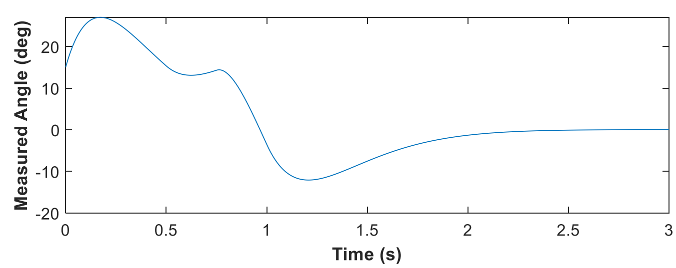


Figure 4: *The input desired angle (left) and resulting system response (right). The simulated response confirms closed loop design characteristics.*

The closed loop system performed exceptionally well. Shown in Figure 5, the steady state behavior of the device-maintained balance within 0.3° of the desired vertical position. The device has significant disturbance rejection capability and could withstand significant pushes or pulls on the frame. Figure 5 shows a 5° disturbance, which settles to within a degree after 1.3 seconds, and has a maximum overshoot of 1.6°. These response characteristics do not quite meet the design specifications; this is likely due to the several modeling assumptions which were made in the design of the controller (negligible damping, negligible motor dynamics due to ODrive torque controller, negligible effects of discrete microcontroller control). However, the device is exceptionally stable, which was the original design intent, so the design specifications have successfully served their purpose in the design of a stable system.

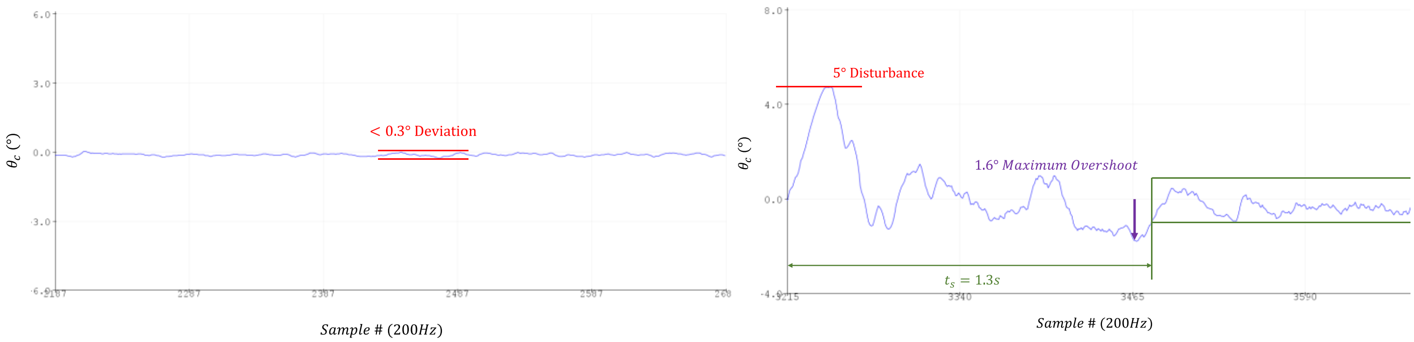


Figure 5: *The controlled response of the cube, in steady state (left) and when rejecting a disturbance (right).*

One note on performance is that because the wheel velocity is not included as a state, the reaction wheel's speed will increase to saturation if the system is perturbed in one direction several times. This could be improved by including the reaction wheel speed in the model and driving this to zero. However, if damping is assumed to be negligible, the resulting state space system has the form:

$$\begin{bmatrix} \dot{\theta}_c \\ \ddot{\theta}_c \\ \dot{\omega}_w \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ a & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} \theta_c \\ \dot{\theta}_c \\ \omega_w \end{bmatrix} + \begin{bmatrix} 0 \\ b \\ -c \end{bmatrix} T$$

This system is in fact not controllable, indicating that the coupling between the wheel and the device due to damping is necessary to model (and model accurately) to control both the wheel speed and the cube angle. If damping is considered, the system is controllable. However, an analysis of the system damping showed it to be quite nonlinear: by running the motor up to some speed, and letting it coast to a stop, linear damping would generate an exponential decay time response. The observed response is very linear in time, indicating a constant damping force, rather than one which is proportional to speed. This function is not linear (it is discontinuous about the origin) and thus is somewhat difficult to model in a linear system. Future work could linearize the system about each side of the origin, characterize the damping fully in these regions, and use this damping model to control the reaction wheel speed.

Conclusion

Over the course of this project, the team constructed a reaction-wheel controlled cube capable of balancing on a single edge. State space control was implemented effectively, and consistent results were achieved. Improvements can be made to the system; a persistent issue was the motor reaching saturation after being disturbed in the same direction multiple times, as discussed in the Results section. Future work would follow as described above. Furthermore, an original goal of the project, standing up from a side using a braking system, was not achieved. If time allowed, implementation of this addition would add value to the project.

Acknowledgements

The team would like to thank Professor Mazumdar for her guidance with this project, as well as the Flowers Invention Studio for providing resources and equipment.

References

[1] M. Gajan and M. Waibel, “Cubli – a cube that can jump up, balance, and walk across your desk,” Robohub, 20-Dec-2013. [Online]. Available: https://robohub.org/swiss-robots-cubli-a-cube-that-can-jump-up-balance-and-walk-across-your-desk/. [Accessed: 22-Apr-2022].

[2] Liao T-L, Chen S-J, Chiu C-C, Yan J-J. “Nonlinear Dynamics and Control of a Cube Robot.” *Mathematics*. 8(10):1840, 2020 https://doi.org/10.3390/math8101840

[3] P.-J. Van de Maele, “Reading a IMU without Kalman: The complementary filter,” Creativity in Automation. [Online]. Available: https://www.pieter-jan.com/node/11. [Accessed: 18-Apr-2022].

ME4012 Grading Rubric for Final Project Report

Team/Project Name: **Cue-B**

Team Members: **Will Compton, Domenic DiCarlo, Thalmus McDowell, Micah Morris**

Metrics	Pts	Comments
Design or Simulation Assumptions (5 pts) *Is the design of the system well-described in the text? *What is the quality of the design, is it well thought out and executed well?		
Model (30 pts) *Is the model section well-written and easy to follow? Is the model error-free? *Does the model accurately capture the relevant dynamics? *Are the equations of motion correctly derived? *Does the team do a good job of estimating the parameters of their plant? *Is there a table of derived/estimated parameters?		
Controller Design/Implementation (30 pts) *Is the controller section well-written and easy to follow? *Does the controller design make sense? *Are they able to meet their desired design metric, achieve disturbance rejection, or achieve stability (for unstable systems)? *Does the team do a good job of implementing a working controller (Labview, Arduino, or Matlab)? *If the controller does not work, can they describe why and what they would change?		
Experimental or Simulation Results (15 pts) *Are the results easy to follow and logically written? *Does the performance of the system match expectations from theory (compare nonlinear simulation with linear model, or compare model with experiment)? If not, did the team explain why? *Are there graphs/figures to show performance?		
Final Paper Quality (20 pts) *Is the abstract/introduction/conclusion well written? *Are the figures captioned and readable? *Did the team make the figures/formatting look nice? *Are there references and are they correctly cited? *Is the paper error-free in terms of spelling and grammar? *Is the paper easy to read and easy to follow logically? *Did the team pick an appropriate level of difficulty for their team size? Where they able to implement what they set out to do? *Did the team finish construction and have a mostly working system? *Does the system work well enough for control?		