Brit Wylie

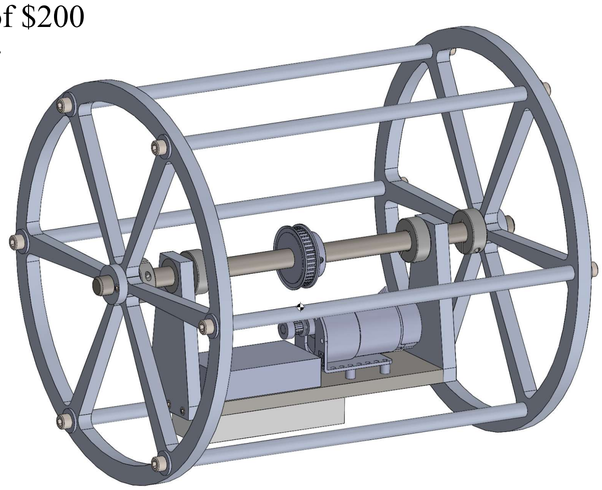
California Institute of Technology
B.S. Mechanical Engineering
bwylie@caltech.edu
(267) 664-3736

Mechanical Engineering Design Portfolio

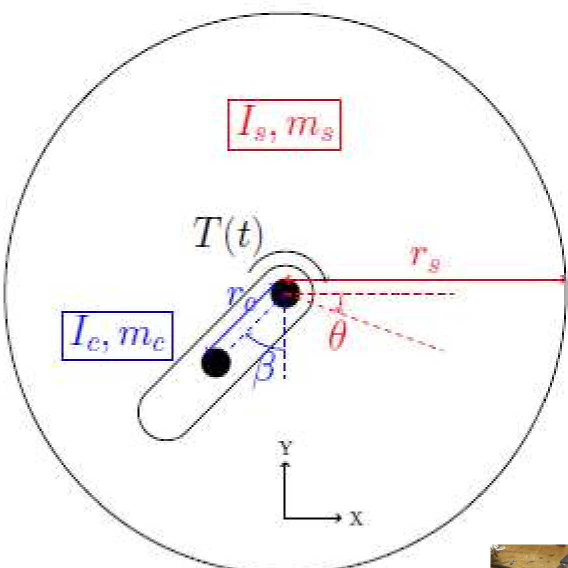
See more at http://britwylie.com

Self-balancing Spool Robot (Spot)

This spoolbot, named "Spot", rolls by shifting its center of mass. It was a project for ME14 and was completed in 3.5 weeks under a budget of $200 with a team of four. Unlike self-balancing robots, which we got our inspiration from, Spot generates movement and balances internally.



The initial goals were:
•Maintain position on 15-degree slope
•Moving on level and angled surfaces
•PID control loop
•Machine a well-toleranced housing
•Successfully tune a responsive control loop
•Gain experience in assembling and tensioning belt drive assembly



I designed the counterweight and determined motor gear ratio using an analysis based on the goals and the system dynamics outlined to the left. This also determined the pulley ratio I also helped to redesign the 3D printed mounts which were used to save cost on the Pololu 50:1 gearmotor. We also planned to modify the HTD pulleys to save cost but one broke in the process and we used to CNC mill to make a new one.

I_s, m_s = mass and inertia of spool
I_c, m_c = mass and inertia of counterweight
r_c = radius from COM spool to COM counterweight
r_s = radius from COM spool to outer edge
θ = angle of spool (CW+)
β = angle of counterweight (CW+)
$T(t)$ = torque from motor on center of spool

$$\frac{d}{dt}\begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & \frac{-b(I_c+m_c r_c^2)}{(m_c r_c^2)(I_s+m_s r_s^2)+I_c(I_s+(m_c+m_s)r_s^2)} & \frac{g m_c^2 r_c^2 r_s}{(m_c r_c^2)(I_s+m_s r_s^2)+I_c(I_s+(m_c+m_s)r_s^2)} & \frac{-b m_c r_c r_s}{(m_c r_c^2)(I_s+m_s r_s^2)+I_c(I_s+(m_c+m_s)r_s^2)} \\ 0 & 0 & 0 & 1 \\ 0 & \frac{b m_c r_c r_s}{(m_c r_c^2)(I_s+m_s r_s^2)+I_c(I_s+(m_c+m_s)r_s^2)} & \frac{-g m_c r_c(I_s+(m_c+m_s)r_s^2)}{(m_c r_c^2)(I_s+m_s r_s^2)+I_c(I_s+(m_c+m_s)r_s^2)} & \frac{b(m_c+m_s)r_s^2}{(m_c r_c^2)(I_s+m_s r_s^2)+I_c(I_s+(m_c+m_s)r_s^2)} \end{bmatrix}\begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{I_c+m_c r_c^2}{(m_c r_c^2)(I_s+m_s r_s^2)+I_c(I_s+(m_c+m_s)r_s^2)} \\ 0 \\ \frac{m_c r_c r_s}{(m_c r_c^2)(I_s+m_s r_s^2)+I_c(I_s+(m_c+m_s)r_s^2)} \end{bmatrix} T(t)$$

$$Y(t) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}\begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix}$$

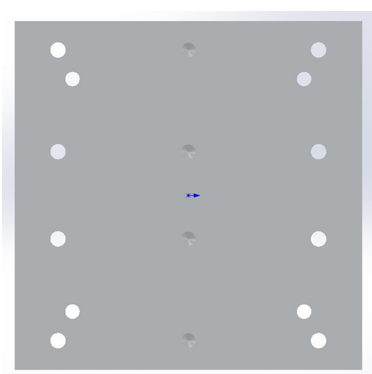
Gear Transmission

Completed in 3 weeks with a budget of \$175 to transmit power between a 10V power supply and a bicycle wheel. Scoring completed according to

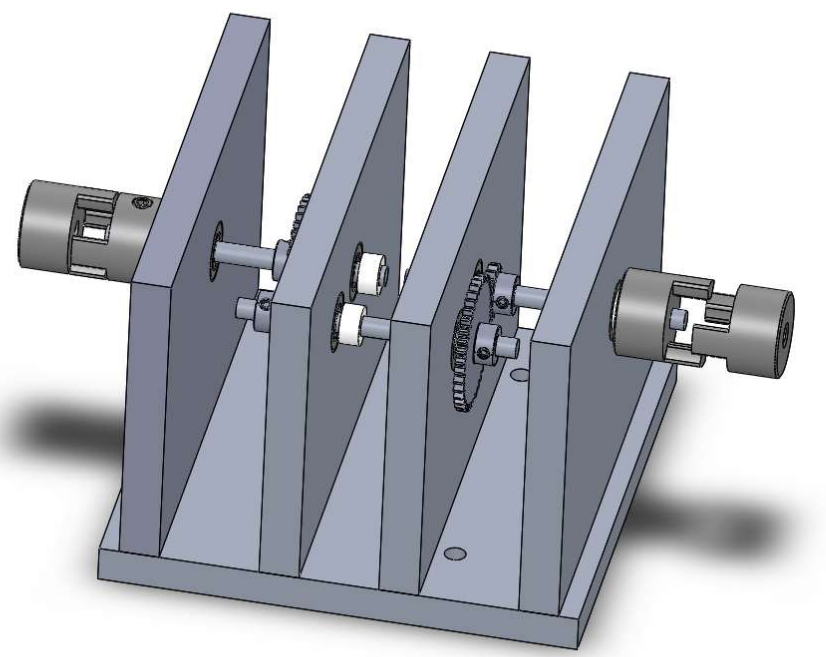
$$S = \frac{\dot{\theta}_{max}}{T_{250}},$$

the cost function of maximum rotational speed of the wheel divided by time to reach 250 rpm. This transmission was made focusing on tolerances, design reliability, and test repeatability. The dowel pin was a slip fit held in place with Delrin collars and bearings. Our largest misalignment was 0.007" of two holes for an internal gear shaft.

We did an assembly-style manufacturing scheme and I drilled the holes and faced some of the walls. I lathed and drilled the Delrin shaft collars to a press fit. I also used metrology to verify our shaft alignment and mounting tolerances.

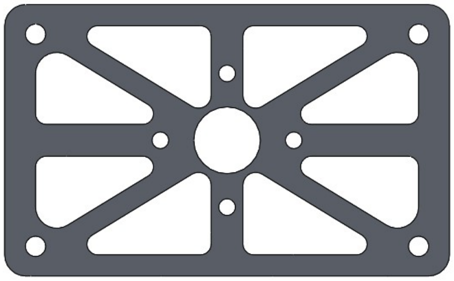


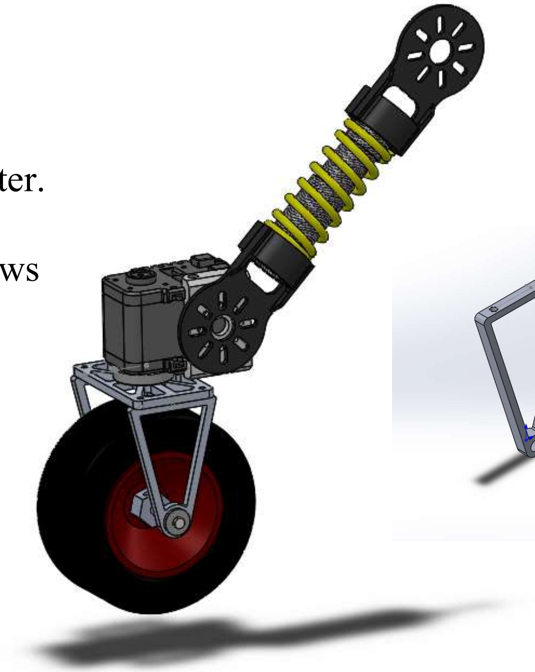
The baseplate (left) had to be modified in the 48 hours before the competition as the geometry of the mounting holes was changed suddenly. Older holes are visible in the top-down view.

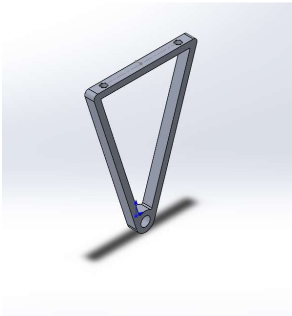


Drive-o-copter

Designed waterjet aluminum mounts to replace 3D printed mounts for an autonomous driving quadcopter. Metal parts could be waterjet quickly in 3 pieces, milled, and fit together with vibration-resistant screws so that many wheel sizes could be tested.





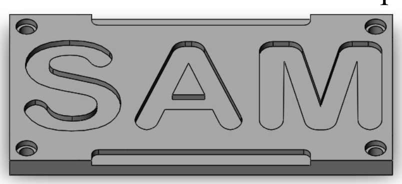


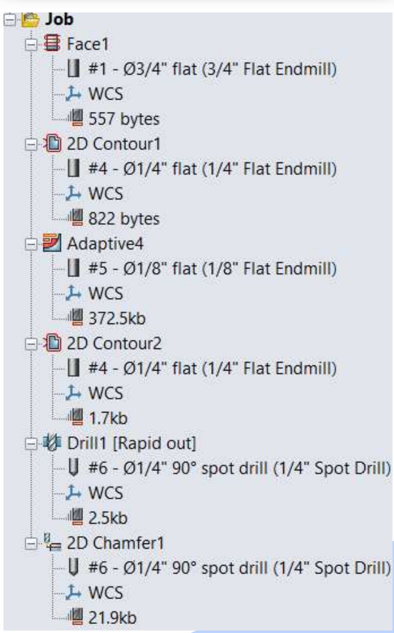
Joystick node for Dynamixel AX-12A in ROS for manual testing.

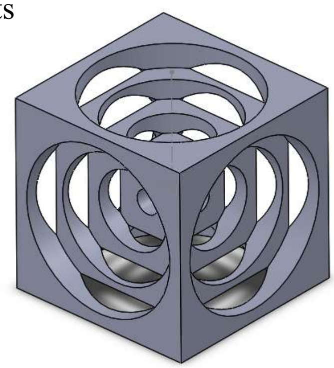
16.99g 3D printed, 13.35g machined

CNC Machining

I modeled and programmed the HAAS with HSMWorks for all of these parts

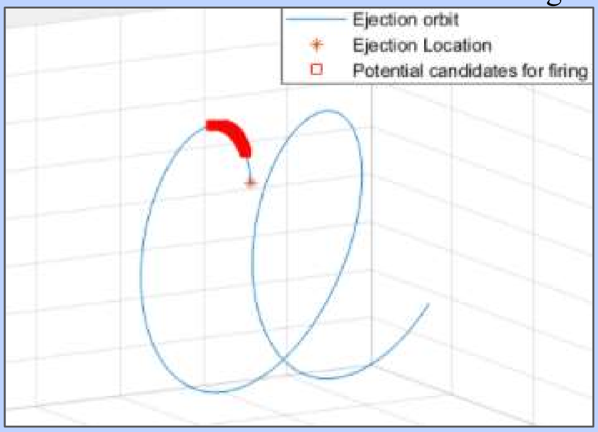


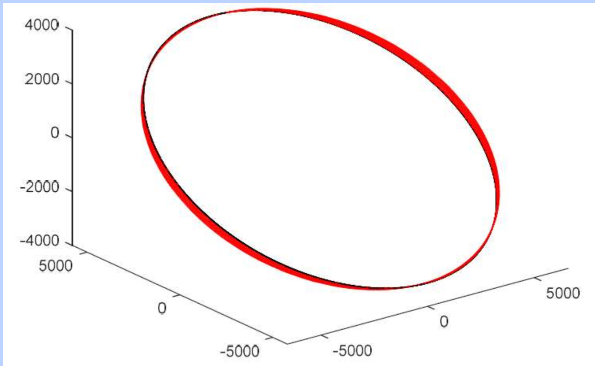




Not pictured: Dash-8 JIC for custom valve interface, CNC in 2 hours 45 minutes from Delrin

PRO formation, automated to test injection conditions:
I developed the code to test ejection conditions from Nanoracks and generate hundreds of firing candidates to optimize delta V usage



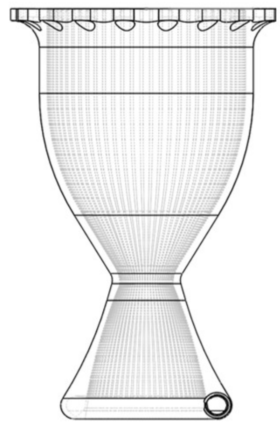


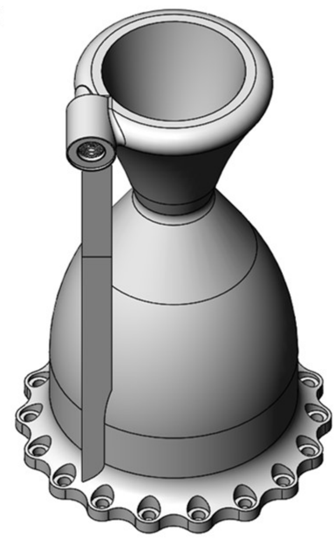
Two-body procession (red) with J2, programmed for the Orbital Mechanics course from scratch in MATLAB, 24 hours at 200 km

AE105 Orbital Mechanics

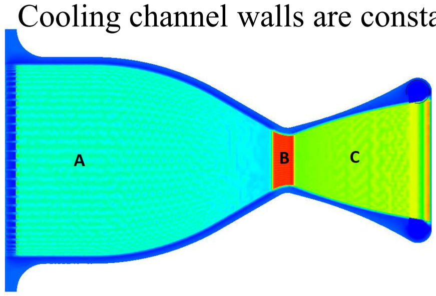
PARSEC - Regenerative Cooling Chamber Design

The rocketry team (PARSEC) at Caltech is designing a Methyl-LOX rocket for the FAR-MARS competition. The combustion chamber must maintain 550 psi, endure a peak temperature of 3400 K, and produce 800 lbf of thrust for 12 s. I used a outer profile generated with RPA to design the combustion chamber with regenerative cooling channels.



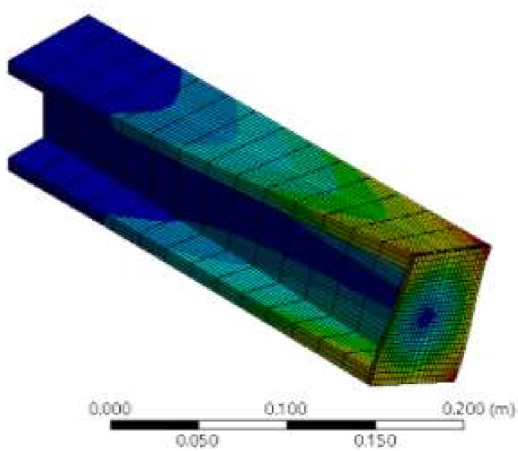


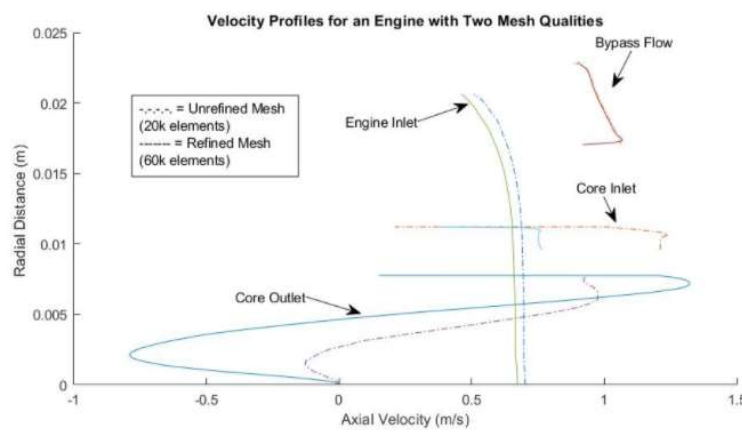
The chamber is designed to be metal 3D-printed using the GALCIT DMLS printer and Inconel 625 powder. It is 0.5mm thick on the inner wall with a 1.7x safety factor. Cooling channel walls are constant width.



A simple flow sim, pressure sim, and thermal sim were used to verify the throat capabilities and an iterative approach was used with these simulations and Bartz cooling calculations in MATLAB

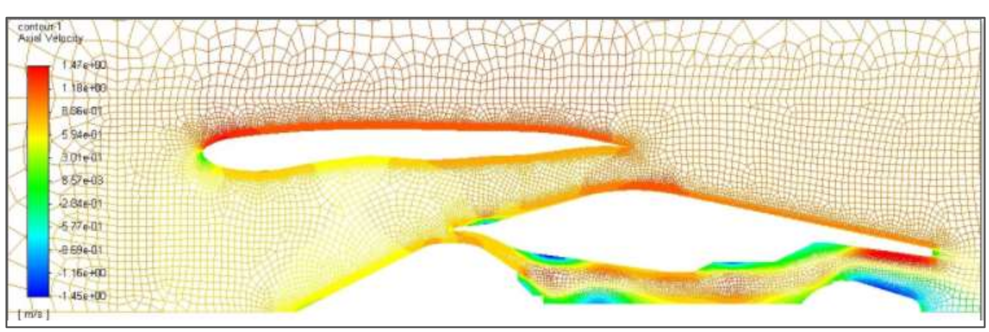
ME50 – Finite Element Analysis and Fluent





Top Right: axial velocity at different mesh sizes to analyze boundary layer; Bottom Right: Fluent axial velocity analysis of a given aircraft engine model

Left: magnitude of deformation of an I-beam under end load and moment constraints, paired with vibrational analysis



DXF Modification and Laser Art Design

Waterjet

PCB Art

Custom DXF

Woodburning

Glass Casting

3D printed, programmed electric forge, melted glass into epoxy base for a class project