DEVELOPMENT OF A FULLY REUSABLE AND AUTONOMOUSLY LANDING SUBORBITAL LAUNCH VEHICLE

Ryan Westcott
Oregon Episcopal School
6300 SW Nicol Rd, Portland, OR 97223

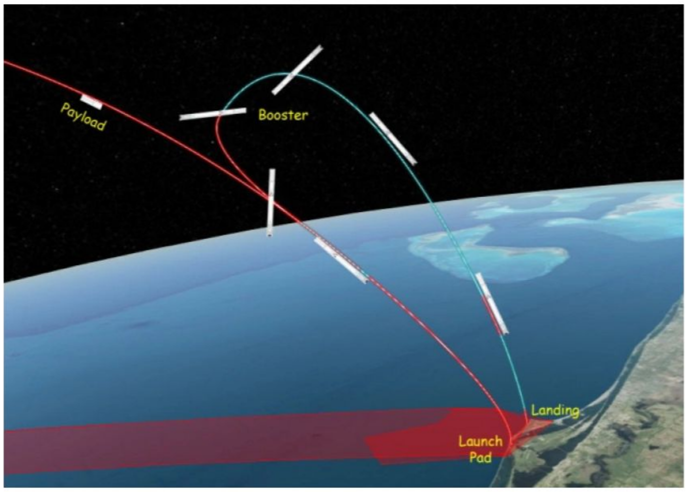
Table of contents:

Introduction

Space exploration is currently experiencing a wave of massive commercial growth. Companies such as SpaceX, Blue Origin, and Rocket Lab are beginning to capitalize on the incredible opportunities that space has to offer: economic growth, scientific discoveries, technological advancements, and many more (Donohue, 2018). These companies, among others, have ignited the world's second space race. This new space race, however, is now between companies backed by billionaires rather than governments backed by politicians. The very nature of this race has pushed these companies to develop more economically viable methods of accessing space. If a government organization is attempting to develop a space launch vehicle, they will spend as much as their budget allows – and maybe even a little more. But private companies need to be able to make money to survive (Fox, 2010). This demand for cost effective ways of accessing space has been a catalyst for reusable rocket development. Nearly every major space launch company today is developing some form of reusable technology, compared to NASA which is still developing expendable rockets (Mosher, 2018).

Reusable launch vehicles look and function exactly like their disposal counterparts between liftoff and delivering their payload into the target orbit. After their job is done,

Figure 1: Example Flight Path of a Reusable Booster



(Jim, 2017)

disposable boosters deorbit and typically burn up in the atmosphere or fall into the ocean. Reusable boosters use retropropulsion methods – along with other control systems – to steer themselves back to the landing zone (O'Connell, 2018). After falling through the atmosphere and being controlled by aerodynamic and active guidance systems, the same engines used to lift the booster and payload up are ignited again to slow the booster down. Advanced stabilization algorithms are used to stabilize the booster so it can touch down softly on the landing zone (see Figure 1). After just a small amount of refurbishment and refueling, the reusable boosters are ready to fly again (O'Connell, 2018).

At the same time that reusable rocket technology is being developed, we are also experiencing a massive growth in technological miniaturization. Not only are our computers, microprocessors, and phones getting smaller, but our satellites and testing equipment are as well. This age of technological miniaturization has led to a large demand for smaller launch vehicles – rockets which are capable of launching smaller payloads into space on a more regular schedule (Foust, 2019). The most recent SpaceX Starlink launch shows how we are now capable of launching 60 communication satellites with the same class of rocket that used to only be able to launch 1 communication satellite – all thanks to technological miniaturization (Lawler, 2019).

Figure 2: Reusable Rocket Preparing to Land

This particular rocket is a SpaceX Falcon 9 which has just returned from space after delivering a satellite into orbit. (SpaceX, 2017)

The issue is that the reusable rocket technology being developed today is only being applied to very large boosters. But as technology continues to decrease in size, there will be a demand for *smaller* and more cost effective, thus *reusable*, launch vehicles. In order to address the demand for smaller and reusable launch vehicles, I worked towards developing a proof-of-concept design for a small launch vehicle with reusable capabilities.

The main challenge in developing a reusable launch vehicle is the development of the propulsive landing system which enables the vehicle to come back to an exact landing site and land in a manner which does not harm the structural integrity (See Figure 2). This project addresses this need, by focusing on developing the hardware, software, and algorithms necessary to achieve a propulsive landing. To accomplish this goal, multiple subsystems and components needed to be developed: 1) a propulsion system that enables the thrust and direction of solid-fueled rocket motors to be controlled, 2) electrical hardware the enables data to be processed, signals to be converted, and power to be managed and transmitted, 3) software to autonomously control the vehicle in real-time, 4) simulation software to optimize the stabilization and throttling algorithm coefficients, 5) a vehicle on which to test my system, and 6) a testing platform to validate the system's performance. Because it would be exceedingly expensive, and well beyond the scope of this project to develop these components for a full-scale rocket, I developed a smaller model as a proof-of-concept.

The propulsion system which I created was 3D printed and enabled pitch and yaw control of the vehicle. The software I developed enabled the vehicle to attempt to stabilize itself during descent using parameters that were calculated using simulation software. The electrical hardware provided a processor on which the software could run, allowed data to be gathered, and enabled the propulsion system to update control parameters accordingly. Finally, a test vehicle was developed which could be dropped from a drone to test how well my system functioned. In all, this project looked towards the future of the space launch industry, identified a need, and worked towards developing a fully reusable launch vehicle in order to address the given need.

Method

The project began with the intention to develop a propulsion system from the ground up: propellant manufacturing, nozzle design, overall motor design, vectoring hardware, and all controlling software and electronics. However, it soon became apparent that building everything from scratch would take too much time and too many resources. After this realization, a large pivot was made to utilize commercially-produced motors instead of developing my own from scratch. Nevertheless, here is a short method on how I worked towards developing my own rocket motors before moving onto the commercially-sourced motors:

The project began by developing a simple solid fueled rocket motor which ran on KNO_3 (Potassium Nitrate) and disaccharides (fructose and sucrose). The nozzle for the motor was designed in the Fusion 360 Computer Aided Design (CAD) program with a converging-diverging structure. The nozzle was then 3D printed with Acrylonitrile butadiene styrene (ABS) filament material. The manufactured nozzle inserted and adhered with cyanoacrylate (CA) into a 3.2cm in diameter polyvinyl chloride (PVC) frame. The first fuel mixture used 65% KNO_3 (as an oxidizer) and 35% sucrose (as fuel). 13 grams of KNO_3 was added to 7 grams of sucrose, and then melted together over a hot plate set to 300ºC. Observations were made on how well the mixture of fuel could be handled and cast into the shape of the final fuel grain. It was noted that the 65:35 KNO_3 to sucrose fuel was too brittle to be easily formed post-heating. A new fuel mixture was created to solve this issue which consisted of 62% KNO_3, 17% sucrose, and 21% fructose. 30 grams of the new fuel mixture were created following the same procedure again, poured into the motor's combustion chamber, and left to cool for 1 hour. After cooling, a 6mm core was drilled using a drill press.

Figure 3: Static Fire Test Stand In Action

Motor mount in the middle, TTS computer to right, plywood base on top of 2”x4” spacers. (Ryan Westcott)

In order to calculate the net thrust of the motor, I needed to develop an apparatus to record the thrust over time (more commonly known as the motor's “thrust curve”). I built an apparatus to achieve this by mounting a 10 Kg load cell to a base, and creating a mount for the motors on top of the load cell (See figure 3). Software was developed to run on a 32 bit Cortex M4 chip using the Arduino language and compiler. This software would read the output of the load cell and log the value to an SD card.

After conducting multiple thrust tests of custom-built motors, it became apparent that designing, developing, and manufacturing my own rocket motors from scratch was not worth the time or effort. The situation was evaluated, and the decision was made to outsource the motor development by purchasing commercially developed rocket motors which enabled me to focus on the more important aspects of this project.

The project really started to take off after moving to actual development of the launch vehicle. After measuring a number of characteristics of the commercially-sourced rocket motor using the TTS, development on the first iteration of the launch vehicle began. The idea with the first launch vehicle was to use gravity and fins to keep the vehicle upright on descent. First, the program OpenRocket was used to design the vehicle's structural components and analyze the stability margins. The structural components were then designed in Fusion 360 and 3D printed in Polylactic acid (PLA). The flight computer was developed with a 32 bit 180 MHz ARM Cortex-M4 processor broken out on a Teensy 3.6 development board. A BMP280 pressure sensor and two SRC-05VDC-SH relays were connected to the M4. Finally, a lithium polymer battery was connected to this circuit through a 1.7V-5.5V step up converter to 5V. The flight computer and other electrical equipment were integrated into the airframe. The flight control software was then developed in C++ using the Arduino compiler language to perform three main functions: 1) send the command to the UAV (Unmanned Aerial Vehicle) to drop the rocket once the target release altitude has been reached 2) after a certain amount of time in freefall, ignite the retropropulsion motors, and 3) constantly log data to the on-board memory. The software was uploaded to the flight computer.

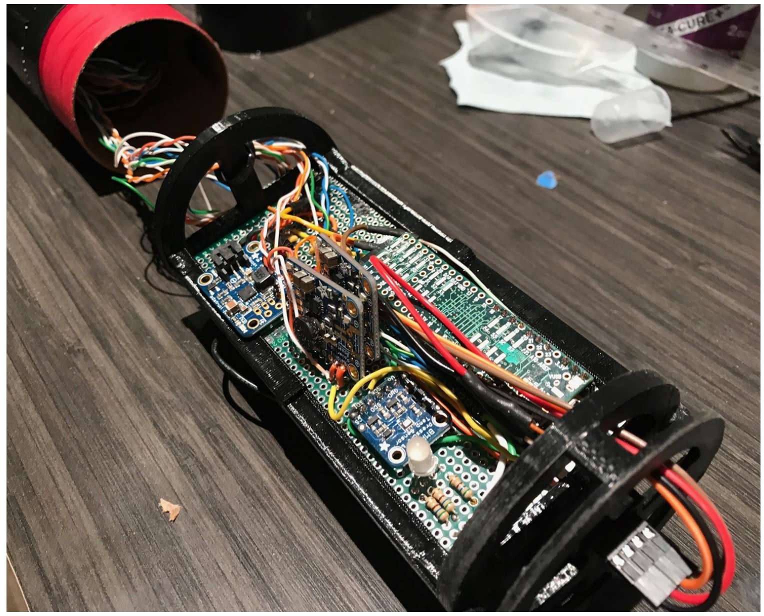
Before testing could start, the drop altitude and ignition delay needed to be calculated. I created a simple physics simulation in a spreadsheet that modeled the flight of the vehicle. The two variables were determined by plugging in different values to the simulation and determining where the vehicle would reach a velocity of 0m/s upon impact with the ground. Once these values had been determined, they were uploaded to the launch vehicle flight computer.

Figure 4: Progression of Launch Vehicle 1 Drop Tests

Left: photo just after releasing from drone. Center: retroburn initiated. Right: Burn continues before impact with ground. (Ryan Westcott)

In order to test just the landing phase of flight, a mount was created which allowed a UAV to drop the vehicle from an exact height. Using this mount, three drop tests were carried out. After the first drop test, the time where the vehicle's velocity was the slowest was calculated. On the two subsequent drop tests, the drop altitude was adjusted so that the launch vehicle would impact with the ground at its slowest velocity.

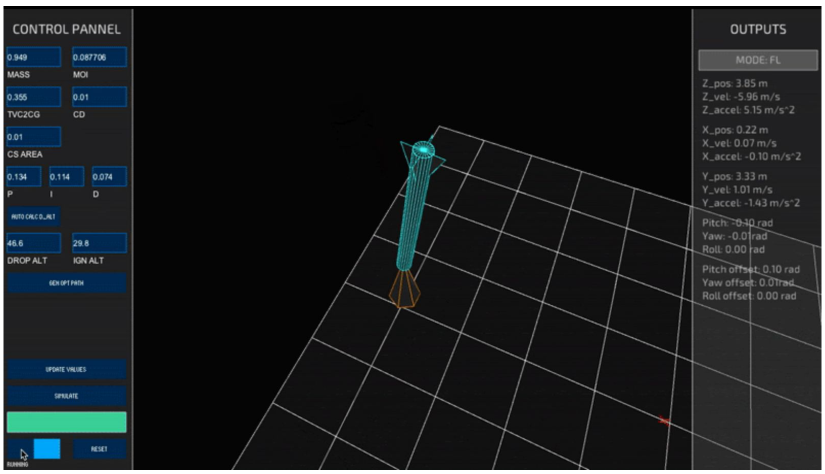
After these initial drop tests (figure 4), it became apparent that dynamic stabilization of the vehicle would not be possible. It turned out that I fell for what's known as the rocket pendulum fallacy. To address this, I designed and developed a novel 3D printed propulsion system which can provide pitch, yaw, roll, and throttle control to the vehicle. This worked by attaching four solid-fueled motors around the circumference of the airframe. Each motor vectored in one dimension tangent to the circumference of the airframe. To apply pitch or yaw to the vehicle, opposite motors were actuated to provide the desired force. To apply roll to the



vehicle, all motors vectored in the same absolute direction to provide the desired force. To throttle down the system, each set of adjacent motors vectored towards each other to cancel out the horizontal thrust created, and reduce the vertical thrust.

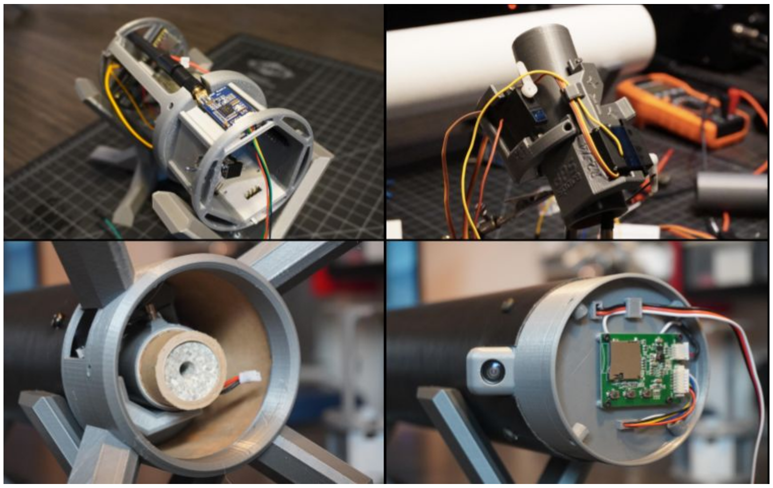
A new flight computer was built on a printed circuit board (PCB) and a 9 degree of freedom (9DOF) inertial measurement unit (IMU) was added. The software was modified to utilize a proportional-integral-derivative (PID) control algorithm to control the angles of the motors. Instead of dropping this vehicle from a high altitude, the vehicle was suspended 3.5m from a hopper test stand. The hopper test was performed by loading the vehicle with fuel, connecting the motor's ignition port to the flight computer ignition ports, hanging the vehicle from the hopper test stand, initiating the flight computer, and sending the command to begin the test.

Due to issues with the hopper test, a hold-down test was devised to more accurately test the stabilization system. A gimbaling mount was designed to enable the launch vehicle to be held down while still moving about its center of mass. The launch vehicle was redesigned to support the new testing configuration. In order to calculate the MOI of the vehicle, a bifilar pendulum test was carried out. A 1” by 6” board was mounted 3m off the ground, with 2m long strings running from the board to the rocket horizontally in flight configuration. The rocket was rotated and allowed to swing around its center of mass. The time it took to perform 10 periods was recorded. This data was used with the bifilar pendulum equation to determine the vehicle’s MOI. This data could then be used to tune the PID gains.



In order to determine the optimal PID gains, a A transfer-function-based PID optimization algorithm was applied to a 3D simulation which was created using MATLAB, Simulink, and SimScape software packages. To perform the hold-down tests, the PID coefficients were loaded onto the flight computer, the launch vehicle was secured to the gimal, the flight computer was powered, the software initiated, IMU calibrated, and the command was sent to ignite the propulsion system. These trials were repeated three times.

After the hold down tests gathered data on the stability performance of the system, drop tests were performed. For these drop tests, aluminum landing gear were developed and dynamic stabilization hardware (stability margin brought to 1.95 calibers) was added to the vehicle. Another bifilar pendulum test took place, the new MOI calculated, and these values were used to



simulate new PID gains. These new PID gains were loaded onto the vehicle's flight computer. A drop test was performed by loading the launch vehicle with fuel, performing ignition system initiation, powering up the drone system, connecting the launch vehicle to the drone via the PWM umbilical cable and suspension line, calibrating the launch vehicle's flight computer, and flying the drone to the predefined altitude. Once the drone reached the target altitude, the launch vehicle would release itself and initiate the stabilization algorithm. Once the rocket had reached the ground, the vehicle would be allowed two minutes to depressurize and cool down. At this point, the flight computer could be powered down and the internal storage card could be removed for further data analysis.

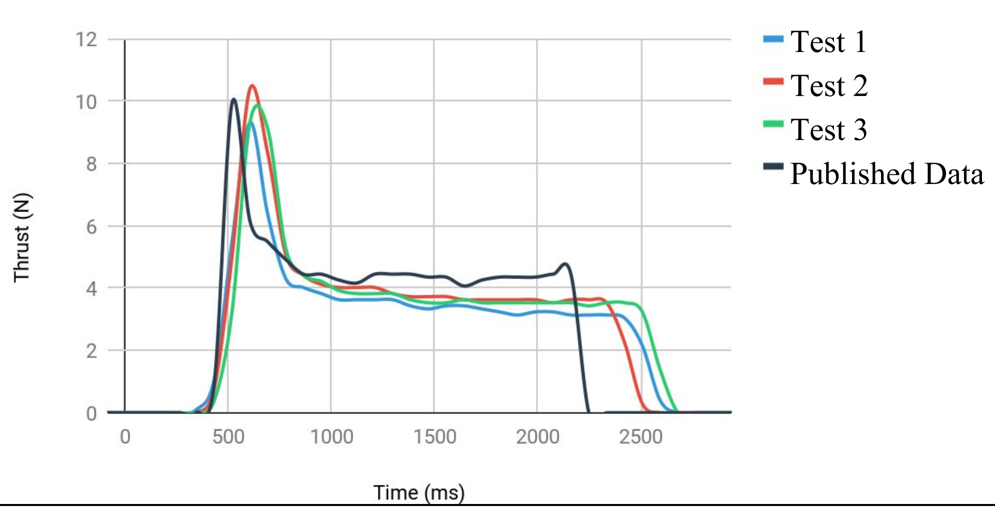
At this stage, a major problem was identified with the custom-developed 3D printed propulsion system.

Because this system utilized four separate motors, it was difficult for the flight computer to ignite each motor at exactly the same moment. A design for a two-axis thrust vectoring mount which uses just one single motor was 3D printed and integrated into a new vehicle. Moving to this vehicle, the ability to throttle the propulsion system was lost. However, since throttling is necessary to achieve a soft landing, an algorithm was invented to achieve this. I call it the "Bimodal Cosine Throttling" (BCT) algorithm. It works by changing the PID setpoint in the pitch and yaw controllers to convert part of the vertical thrust component into a horizontal thrust component. Simulations show that the flight envelope where the retropropulsion system ignites can successfully be increased from a point to an area.

Results

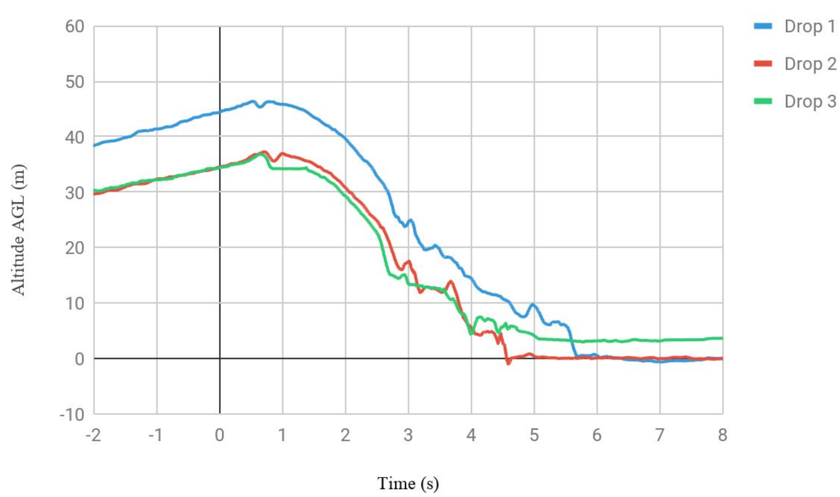
After the decision was made to pivot to commercially sourced motors, the thrust curve needed to be determined in relation to the beginning of the ignition signal. On average, thrust began to be produced 0.401 s after the ignition signal was sent (±13.965% with a standard deviation of 0.040 s, see Figure 7). This data turned out to be extremely important as the published data varied significantly from the recorded data. The published data showed that the thrust terminated 0.24 seconds before the average thrust from the actual tested motors. Each motor test consisted of 35 data points.

Figure 7: Thrust Curve of Commercially-Sourced Estes C6 Rocket Motor



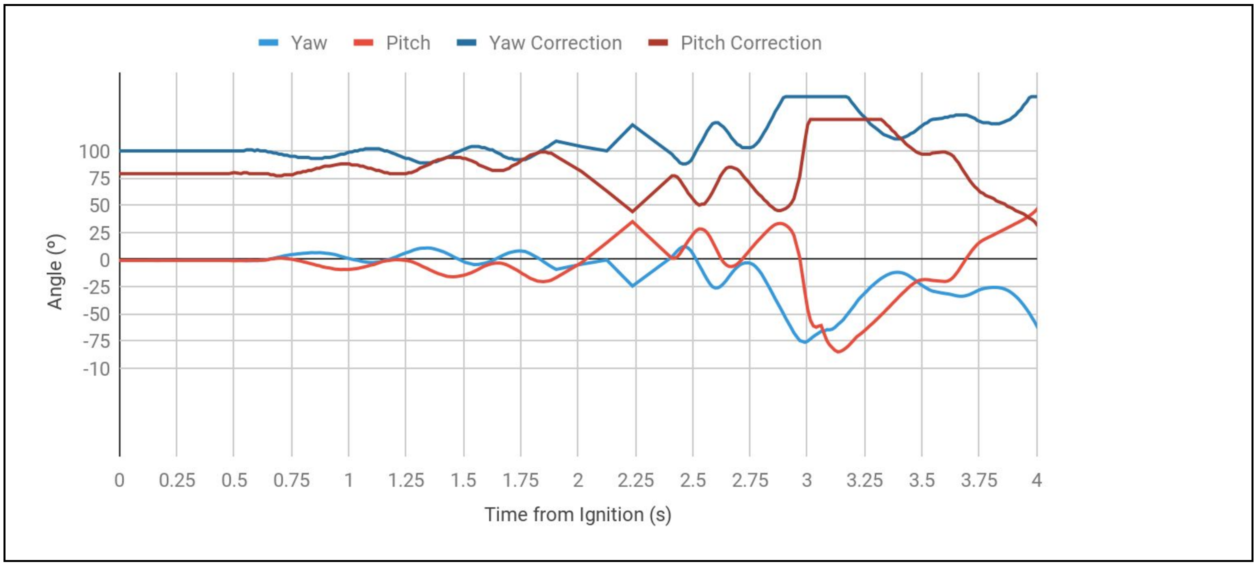
After the individual motor performance had been recorded and the first version of the launch vehicle developed, three drop tests were performed. The first test had the target of achieving the slowest velocity 10 m above ground level. The two subsequent tests were dropped from a lower height in order to attempt to have the slowest velocity occur when the vehicle touches down on the ground, see Figure 8. Unfortunately, the instability of the vehicle in flight caused significant pressure differences around the avionics. These pressure differences caused noisy altitude data, in turn preventing this data to be analysed (e.g. you may notice points where it looks like the vehicle is gaining altitude, which is impossible).

Figure 8: Drop Test Altitude Profile of Launch Vehicle Version 1



After rebuilding the launch vehicle to enable active stabilization, the pitch and yaw angles were recorded, as well as how much each axis of the thrust vectoring system deviated from vertical in order to provide a restoring force to the vehicle. Figure 9 shows the orientation of the vehicle during the hopper test, as well as the correction angles on each axis.

Figure 9: Pitch and Yaw vs Pitch and Yaw Correction During Hopper Test with Launch Vehicle Version 2



The final stage of testing in this project were the drop tests from the drone. At each stage, the control gains (P, I, and D values) were updated, as well as the release altitude and ignition delay (see Table 1). These were able to be updated at each drop because they impacted the vehicle independently (e.g. release altitude does not affect the stability margin of the PID loop). The results of how well the stability software worked can be seen in Figures 10 and 11.

Table 1: Subset of Flight Parameters for Each of the Four Drop Tests

	Release Alt (m)	**Ign. delay (ms)**	**P (unitless)**	**I (unitless)**	**D (unitless)**
Drop 1	26	1400	2.0	0.3	0.7
Drop 2	14	1400	1.9	0.3	0.8
Drop 3	12.5	1300	1.5	0.3	0.95
Drop 4	13	1300	1.7	0.3	0.9

Figure 10: Deviation From Vertical for Each Drop Test on LV Version 4

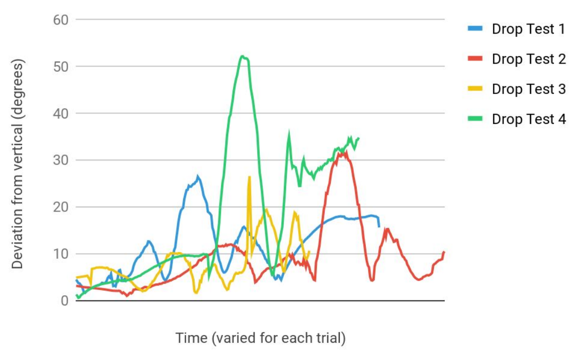
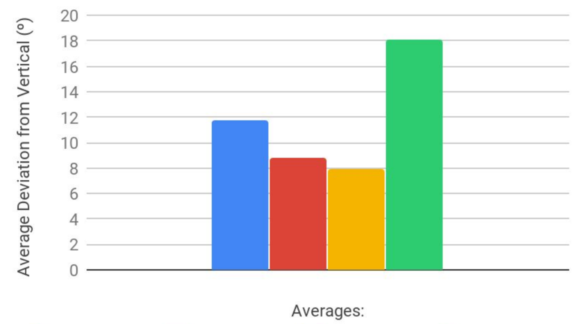


Figure 11: Average Deviation From Vertical for Each Drop Test on LV Version 4



In total, approximately 1.68 million data points were gathered over the course of the final four drop tests, with 56,000 being recorded during the landing phase of flight. These points included data such as pitch, yaw, roll, temperature, pressure, humidity, acceleration (on all three axes), angular velocity (on all three axes), magnetometer values (on all three axes), actuator positions, ignition states, and other flight parameters. Contact Ryan Westcott for raw data.

Figure 12: Landing burn ignition profile without throttling.

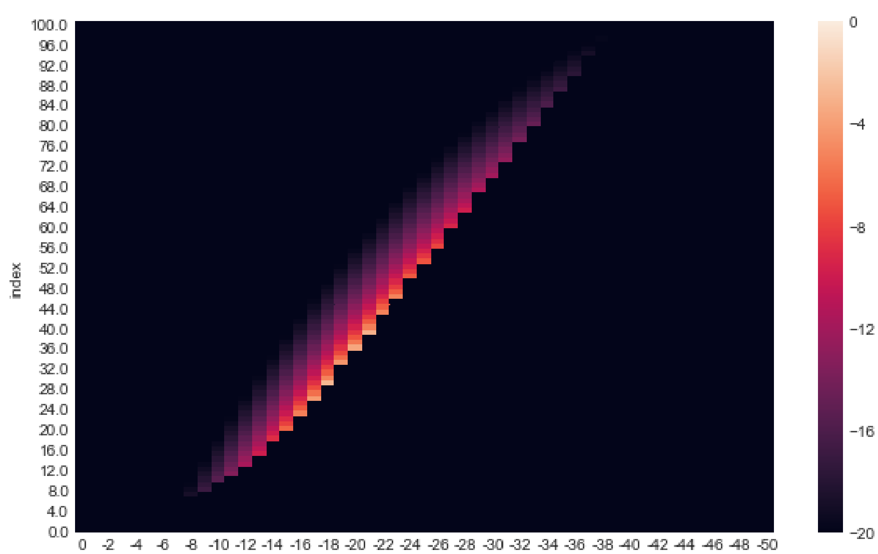
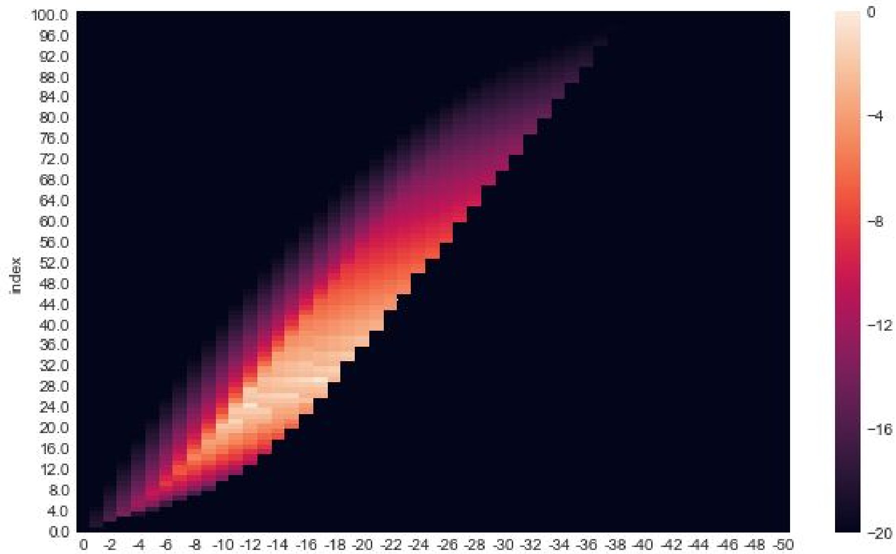


Figure 13: Landing burn ignition with throttling.



The previous two figures show the retropropulsion ignition envelope with and without throttling enabled. The Y-axis is the altitude at retropropulsion motor ignition (m), the X-axis is the vertical velocity at retropropulsion ignition (m/s), and the color represents the impact velocity (m/s). Without Bimodal Cosine Throttling, the vehicle must pass through a very small, very specific flight regime in order to land with a near-0 velocity. However, with the throttling algorithm enabled, the system can be ignited within a much larger flight envelope.

Discussion

The most important aspect of this project was the PID stabilization algorithm which was used to keep the vehicle vertical when landing. As explained earlier, the P (Proportional), I (Integral), and D (Derivative) gains must be simulated in order to determine the optimal values. The P gain looks at the current deviation from vertical on each axis, and changes the control parameters proportionally. A higher P value results in a more responsive system (with more overshoot), while a lower P value results in a less responsive system (but with less overshoot).

The I gain evaluates how long the vehicle has been deviated from vertical, and corrects accordingly. A higher I gain results in more precise long-term stabilization (but more prone to drifting), while a lower I gain is less precise long-term stabilization (but with less drifting). The D gain looks at how fast the vehicle is moving towards or away vertical on each axis. A higher D gain typically results in decreased oscillation amplitudes (but is prone to errors at small angular velocities), while a lower D gain results in increased oscillation amplitudes (but is better at small angular velocities). These three values must be tuned together to create the most stable system possible.

In the first drop test, the P-value was slightly too high and the D-value was slightly too low. This is visible is the large overshoot after each corrective movement by the stabilization system, resulting in an average of 11.8º off of vertical. In order to reduce the overshoot, the P-value was decreased and the D-value was increased for the second drop test. The new values improved the oscillation of the vehicle and resulting in an average of 8.8º off of vertical. The P-value was once again lowered and the D-value raised, this enabled the system to achieve the most precise landing attempt yet, resulting in an average of on 7.9º off of vertical. However, the low P-value caused a loss of control authority at the end of the burn, resulting in a large deviation near the end. Because of this, the P-value was increased slightly for the final landing attempt. Unfortunately, on this last attempt, an anomaly occurred with the motor ignition system preventing enough thrust to be created. This caused asymmetric force to be applied to the vehicle, causing it to have the highest average deviation (18.1º) from vertical of any of the four trials. This also created the large spike in the middle of the test, as seen in figure 10.

Weaknesses were present in certain areas of this project. A BMP280 barometric pressure sensor was used by the flight computer to calculate the altitude of the vehicle for each trial. Barometers are very sensitive pieces of equipment, and any small pressure difference can lead to a large difference in the altitude output. This source of error is visible in the altitude profile of the LV1 drop tests (Figure 8) where it appears that the vehicle goes up and down, where in reality it is small pressure spikes from the propulsion system affecting the altitude reading. Another source of error occurs in the position readings of the stabilization actuators. These values are based on what position value is being sent to the motors, not the actual position. Because of this, we do not know how fast the response of the actuators are, or if they are being affected by outside factors. This could be fixed by adding a sensor to the stabilization system which would enable the flight computer to measure the exact position of each of the actuators.

The most important aspect of this project going forward is the refinement of the PID values and throttling software. If these values are correctly determined, then the apparatus that I developed should be able to achieve the original goal of propulsively landing. In order to achieve this goal, software can be used to simulate the system and identify the optimal values. Researchers at Slovak University of Technology in Bratislava achieved this goal by developing software which ran on the MATLAB platform (Bakošová, Oravec, & Čirka, n.d.). The overall implications of the findings in this project are far-reaching. The PID algorithm which I developed in this project functions similarly to the ones used by researchers at the Pavai College of Technology (Sumathi & Usha, 2014).

The technology developed throughout this project has a significant potential to help advance the development of small reusable launch vehicles. The launch vehicle, as well as the stabilization algorithm, are expandable and future research will be able to determine more precise PID gains. The combination of these improved values, as well as refined hardware, will undoubtedly enable a fully propulsive landing of a small rocket in the near future.

Works Cited

Bakošová, M., Oravec, J., & Čirka, L. (n.d.). *Software for PID Controller Tuning*. Retrieved from Semantic Scholar database. (Accession No. 5551900f5750d06547df92dbd9daa4a05976)

Dey, C., Mudi, R. K., & Lee, T. T. (2006). A PID controller with dynamic set-point weighting. Retrieved from Semantic Scholar database. (Accession No. 10.1109/ICIT.2006.372306)

Donohue, T. J. (2018, December 17). Space: the new economic frontier. Retrieved June 11, 2019, from Space News website: https://spacenews.com/space-the-new-economic-frontier/

Foust, J. (2019, May 8). Commercial, not government, demand will drive size of small launch vehicle market. Retrieved June 11, 2019, from https://spacenews.com/commercial-not-government-demand-will-drive-size-of-small-launch-vehicle-market/

Fox, S. (2010, June 3). 6 Private Companies That Could Launch Humans Into Space. Retrieved June 11, 2019, from https://www.space.com/8541-6-private-companies-launch-humans-space.html

Gstattenbauer, G. J. (2006, March). *Cost Comparison of Expendable, Hybrid, and Reusable Launch Vehicles*. Retrieved from DTIC Online database. (Accession No. a451291)

Klevanski, J. (2002, January). Progress in the Design of a Reusable Launch Vehicle Stage. Retrieved from ResearchGate database. (Accession No. 224783139)

Lawler, R. (2019, May 23). SpaceX just launched a Falcon 9 loaded with Starlink internet satellites. Retrieved June 11, 2019, from https://www.engadget.com/2019/05/23/spacex-starlink-launch/

Mosher, D. (2018, August 10). NASA just gave $44 million to 6 private companies — including Jeff Bezos' Blue Origin — to develop 'tipping point' space technologies. Retrieved June 11, 2019, from https://www.businessinsider.com/nasa-tipping-point-private-space-exploration-funding-2018-8

O'Connell, C. (2018, February 6). Launch, land, repeat - reusable rockets explained. Retrieved June 11, 2019, from https://cosmosmagazine.com/space/launch-land-repeat-reusable-rockets-explained

Sumathi, R., & Usha, M. (2014). Pitch and Yaw Attitude Control of a Rocket Engine Using Hybrid Fuzzy-PID Controller. Retrieved from Semantic Scholar database. (Accession No. 10.2174/1874444301406010029)

Picture Citations

Jim. (2017, May 6). SpaceX | How They Do Booster Landing. Retrieved June 11, 2019, from https://jarphys.wordpress.com/2017/05/06/spacex-how-they-do-booster-landings/

SpaceX. (2017, September 7). More photos from today's Falcon 9 launch and first stage landing. Retrieved June 11, 2019, from https://twitter.com/spacex/status/905861374445850628

Acknowledgements

I would like to thank Walker Jones and Emily Williams for their help as ground support crew throughout testing. They videoed trials, helped carry equipment, and managed onlookers to make my trials quicker, safer, and more efficient.