

2025 International Rocket Engineering Competition Accepted Podium Abstracts

Written by Students
Compiled by the Experimental Sounding Rocketry Association

May 2025



Contents

- 144 Rochester Institute of Technology
Extending OpenRocket with MATLAB for thorough analysis and optimal design
- 146 Royal Melbourne Institute of Technology
Design and Testing of a N-Modular Redundant Flight Computer Architecture
- 148 Sapienza University of Rome
Development and Validation of a Novel Framework for 6DoF Flight Simulations of a Sounding Rocket
- 172 University of Florida
Real-Time Apogee Prediction and Correction in Rocket Flight via Active Drag Modulation
- 190 Virginia Polytechnic Institute
Iteration Upon the Active Drag System A Closed-Loop System for Precision Apogee Control of a Sounding Rocket
- 203 Case Western Reserve University
Design and Manufacturing of Forged Carbon Fiber Flaps
- 204 Chulalongkorn University
Development of a Non-pyrotechnic Recovery System for High-powered Rockets
- 208 Mississippi State University
Applied Machine Learning-Based Additive Manufacturing and Topology Optimization for Structural Aerospace Components
- 212 Politecnico di Torino
FEM Validation of an Additively Manufactured Flange: Onyx vs. PEEK
- 216 The University of Melbourne
Lightweight Neural Compression of Precomputed Flight Configurations for Model-Based Apogee Control Systems in High-Powered Rockets

- 218 The University of Texas at El Paso
Development and Testing of an Innovative Multi-Layer Plate Deployment (MLPD) Mechanism for Precision Altitude Control
- 302 Federal University of ABC
Nose Cone Manufacturing Using Vacuum Lamination Technique
- 306 Kent State University
Oblique Shockwave and Pressure Gradient Analysis of a Von Karmen Nose Cone
- 312 Texas Tech University
Laboratory Analysis of Solid Rocket Motor Nozzle Slag
- 313 Universidade Federal de Juiz de Fora
Machwave: a Stochastic Solid Rocket Motor Simulation Software
- 400 Duke University
Active Roll Stabilization using Canard Control Surfaces
- 401 Federal University of Santa Catarina
Design and Development of a High-Performance Rocket Nozzle with Parabolic Geometry Defined by a Multi-Parameter Rao Method Application and Optimized Structural Integration
- 405 University of Maryland, College Park
Modeling and Design of a Custom Radio for Telemetry and Live Video
- 407 University of Minnesota, Twin Cities
Design, Development, and Testing of a Telemetry Agnostic Ground Station
- 409 University of Washington - Seattle
SRAD Carbon Fiber Composite Airframes
- 601 Technical University of Munich
Design and Process Development of a Multi-Shot CFRP Fincan
- 700 Cornell University
Autonomous Parachute Guidance System
- 701 Indian Institute of Technology Madras
Active Braking and Control System for a Sounding Rocket
- 703 Monash University
From the Ground Up: A Full-Stack Architecture to Propel Rocket Avionics
- 704 The University of Sydney
Candlebark: Hybrid Rocket Motor Simulation Suite for High-Power Student Rocket
- 706 University of Calgary
The Power of Risk Mitigation Through the Use of Software

Extending OpenRocket with MATLAB for thorough analysis and optimal design

Team 144 Project Technical Presentation to the 2025 IREC

Yevgeniy Gorbachev, Ares Bustinza-Nguyen, and Kyle Scher

Rochester Institute of Technology, 1 Lomb Memorial Drive, Rochester, New York, United States of America

INTRODUCTION

Several open-source simulators are used in high-powered rocketry. OpenRocket is one, and its key differentiators are its excellent user interface and integration with a database of thrust curves. It will therefore be our standard for the foreseeable future, at least for initial design and sizing. The tradeoff is that—unlike fully programmatic simulators like RocketPy—automated simulation is not straightforward. Moreover, any extensions to the model calculations must be provided with a graphical interface, consuming more developer time. We present and openly distribute a program interface, inspired by the Python package `orhelper` [1] that uses MATLAB’s robust Java integration to expose all of OpenRocket’s functionality to the MATLAB workspace. This permits programmatic analyses in a language familiar to many engineers while retaining OpenRocket’s key advantages. We present our first use of this integration to numerically optimize the launch vehicle. We develop a MATLAB toolchain to download and analyze National Weather Service upper-air forecasts and present a direct comparison between the forecast systems’ accuracy in predicting the wind vector. We then determine the exact impacts the inclusion of these data has on the simulated trajectory.

VEHICLE COMPONENT OPTIMIZATION

The combination of fin dimensions and ballast weight (usually located in the nose cone) must achieve the correct apogee, adequate stability, and flutter velocity margin to an unsafe level. These parameters are largely free to vary to satisfy these requirements. This represents a multi-input multi-output constrained optimization problem, to which our approach to date was highly manual and tedious. It is relatively straightforward, however, to define optimality numerically; the ability to programmatically test variations lets us study tradeoffs and arrive at optimal solutions more quickly and reliably.

The problem then turns to one of specifying a cost function with desirable properties. Generally, we care that the mentioned metrics fall within certain ranges, but are largely indifferent within; lacking an active control system, the apogee need only be optimized to a tolerance specified by our sensitivity analysis. The static stability margin need only be optimized to fall within the limits specified by the Design, Test, and Evaluation Guide [2]. The fin flutter velocity margin must exceed 50%, and there is little benefit beyond. Additionally, we were forced to work within the size constraints of available carbon fiber plate stock. We defined a cost function that captures these criteria and used MATLAB’s `fminsearch` routine based on the Nelder-Mead Algorithm to optimize the vehicle. Nelder-Mead is a derivative-free local optimizer suitable for low-dimensional optimization [3].

ACCURACY ASSESSMENT OF UPPER-AIR FORECASTS

The stable release of OpenRocket at the time of development (23.09) does not have the ability to describe pressure, temperature, and wind vector profiles. The integration makes the development of extensions to achieve this trivial. While this inclusion is expected to improve the simulation accuracy, the authors are not aware of a direct numerical analysis of the extent to which accuracy can be improved, or of the limits of accuracy of each forecast system. We defined six models to evaluate: three resolutions of the Global Forecast System (GFS), two of the North American Mesoscale (NAM) model, and the High-Resolution Rapid Refresh (HRRR). They have drastically different grid (110 km to 3 km), vertical (500 km to 250 km), and temporal (6 hour to 1 hour) resolutions. Naively, the highest resolution performs the best. However, we demonstrate that this is not necessarily the case.

Our analysis requires a large database of objects drifting in the wind, which we find in SondeHub, a website that aggregates upper-air weather balloon (radiosonde) GPS trajectory logs. To best analyze accuracy, we use all of the radiosonde observations in the continental U.S. from the June of 2024, totaling 2900 trajectories. For each trajectory and each forecast model, we initialize a state at the first GPS packet, evolve its lateral position according to the wind, feed the vertical position in directly, and compare the results. We identify the prediction circular error probable (CEP) against flight time and the wind vector RMS error and bias against ground-level altitude as the key figures-of-merit.

We found that, for all outputs, the model makes significantly more difference than the grid resolution output. Up to a 24 hours of lead time, the wind vector bias is not significant, and the prediction error follows the characteristic of a random walk. In the best case scenario, the 3 km NAM analysis predicted drift to within 250 m at a 3-minute flight time, growing with an approximately square-root relationship with time (as expected of a random-walk). Analyses can provide a data source for assessment of typical conditions during the design process and can aid recovery with only partial trajectory information after launch. However, contrary to expectation, the GFS produced the best forecasts, exhibiting less bias at higher altitudes with commensurably lower prediction error. Therefore, we conclude the NAM is best suited for climatology and the GFS for tuning the launch vehicle during the week of launches.

SENSITIVITY ANALYSIS

We performed parameter sweep and Monte Carlo analyses to evaluate the effects of variation in launch conditions. Analyses evaluated over the month bracketing the launch date (in 2024) provide the typical values and expected variation in the Midland atmosphere, which set the variations used in our sensitivity analyses. To assess the impact of per-altitude atmospheric data, we ran the simulation with the average atmosphere, customizing each model one at a time. The custom atmospheric and wind models do not significantly affect the rocket's simulated ascent. However, the inclusion of a pressure profile enables the geometric altitude to be corrected to achieve the standard-atmosphere indicated altitude against which the team is scored. The pressure/altitude over Midland during June is non-standard but fortunately highly consistent, so we expect to be able to reliably optimize to a new indicated apogee. but the custom wind profile provides a more accurate spread of the recovery location.

A Monte Carlo analysis, uniformly sampling from the expected spread in launch conditions, allows us to determine which factors and interactions most significantly affect the launch. We confirm the intuitive result that wind speed and ground temperature are most impactful to apogee (affecting the rod exit and air density, respectively), and moreover that the interactions between any varied parameters are not statistically significant.

CONCLUSION

We developed and openly released a MATLAB interface to OpenRocket. We began to leverage the capabilities this offers, in numerically optimizing the fins and ballast, integrating forecast data, and performing sensitivity analyses. Moreover, we present an original assessment of the forecast systems' accuracy, deriving proof of which model outputs best apply to different use cases. Our flight at the 2025 IREC will provide a point to validate or refute our new analysis and optimization process.

REFERENCES

- [1] Popescu, A., “orhelper,” 2022. URL <https://github.com/SilentSys/orhelper>.
- [2] ESRA, “Design, Test, & Evaluation Guide,” Standard, Experimental Sounding Rocket Association, October 2023. URL https://www.soundingrocket.org/uploads/9/0/6/4/9064598/2023-sa_cup_dteg_v2.2.9_10-24-23.pdf.
- [3] Lagarias, J. C., Reeds, J. A., Wright, M. H., and Wright, P. E., “Convergence Properties of the Nelder-Mead Simplex Method in Low Dimensions,” *SIAM Journal on Optimization*, 1998. URL https://www.researchgate.net/profile/Jeffrey-Lagarias/publication/216301003_Convergence_Properties_of_the_Nelder--Mead_Simplex_Method_in_Low_Dimensions/links/5400a0230cf23d9765a3f976/Convergence-Properties-of-the-Nelder--Mead-Simplex-Method-in-Low-Dimensions.pdf.

Design and Testing of a N-Modular Redundant Flight Computer Architecture

Team 146 Project Technical Presentation to the 2025 IREC

Hugo Begg and Matthew Ricci
Royal Melbourne Institute of Technology, Melbourne, VIC, 3000

INTRODUCTION

Redundant flight computer systems are critical in ensuring reliable and safe recovery deployment in sounding rockets. The Australis flight-computer architecture utilised aboard RMIT University HIVE's entry to the International Rocket Engineering Competition (IREC) uniquely incorporates N-modular-redundancy¹ (NMR) in a configurable hardware and firmware solution. Commercial-off-the-shelf (COTS) solutions were found to have limited redundancy in their implementations, which often requires additional dissimilar flight computers to facilitate a dual-redundant avionics system. Student avionics development extensively targets a modular avionics system whose core functionality is made up of entirely modular electronic sub-systems connected over communications networks. Solutions like this, while adaptable and scalable, introduce increased points of hardware failure and rarely incorporate redundancy internal to their overall avionics' framework. The Australis avionics architecture addresses redundancy at both a hardware and firmware level, providing an entirely configurable framework to suit many NMR requirements.

HARDWARE DESIGN

The core functionality of Australis' NMR architecture can be reduced primarily into its internal communications design. A single Australis flight computer operates on two STM32F439 microcontrollers each connected to a multi-sensor suite over an isolated serial-peripheral-interface (SPI) bus — The complete NMR solution makes use of *two* Australis flight computers linked by redundant communication protocols. The eight dissimilar devices forming the two sensor suites on any Australis board are made up of a high-resolution barometer and gyroscope and two accelerometers operating at differing scales (tuned to either high or low acceleration events). Data from these sensors are fed into a Kalman filter², benefitting state estimation through additional measurements for statistical error correction to accurately predict the flight vehicle's position in free space. This approach enables the recovery deployment critical capabilities: apogee prediction and altitude determination. Any microcontroller and sensor suite pair can be considered a voting 'candidate' when discussing the NMR system, however a single 'candidate' can be elected as an 'arbiter' for the overall system. The NMR capabilities become apparent when each voting candidate passes its 'votes' regarding critical flight events (namely apogee and main parachute deployment timing) to the arbiter upon which it can act. The arbiter is configured to operate under a reduced load operating in a non-multi-threaded operating environment unlike the voters. This minimises computational workload and general system complexity, reducing the likelihood of internal system faults. Its primary role is to assess candidate votes and act on recovery deployment events based on a majority outcome.

The communications network between the two Australis flight computers are at the heart of its NMR capabilities. Inter-board communications are facilitated by a global controlled-area-network (CAN) to which each voting candidate is linked. Also available (either internal or external to the flight computers) are UART, I²C, an 8-bit GPIO bus, an additional CAN bus to link external sub-systems if required, and a 100mW LoRa transceiver. Whereas primary communications occur over the global CAN bus, redundant communication solutions can be used to pass voting information from candidates to the arbiter. In the event of a candidate system failure, the two remaining candidates can still pass their votes to the arbiter.

The significance of the term 'N-modular redundancy' is inherent to the Australis flight computer hardware architecture. N number of Australis flight computers (of identical function) can be linked to the arbiter to improve the majority voting design accuracy, limited only by power and space requirements in the flight vehicle. The design also supports dual-mode redundancy whereby two voting candidates can operate in parallel and in pseudo isolation not unlike avionics system using two dissimilar flight computers. Alternatively, to implement traditional dual-mode redundant avionics solution, electrically isolated Australis flight computers can be operating on a single voting

candidate of dissimilar sensor suites. In effect, it is entirely up to the flyer's discretion as to how the Australis hardware architecture can be used – it is a modular solution in a single, configurable package.

FIRMWARE DESIGN

Australis' hardware architecture necessitates an equally flexible firmware implementation to enable its development and application in varying configurations. Subsequently, based on FreeRTOS, the Australis firmware sustains a robust 'core' system that supplies the base communication and hardware interfaces, as well as the minimum required logic for flight. The core is responsible for both defining generic interfaces for sensors and hardware components and provides internal communication systems which efficiently pass data from external sources such as CAN or LoRa within a thread-local context. Critical system tasks are also implemented as a part of the firmware core, defining the logical operation of binary data storage, real-time calculation of estimated flight state, and the user-facing shell for interactive firmware testing and system control. Extra functionality may optionally be included by layering system defined and independent modules on top of the core layer. These modules interact laterally to integrate with target-specific source code, enabling critical systems to communicate with minimal coupling. The architectural benefits of the firmware's modular design — which facilitates Australis' capability for numerous flight configurations — are significant; The approach allows Australis to function not only with triple-mode redundancy (TMR), but as an independent flight computer or a larger N-Modular system.

Under the TMR configuration, each voting candidate maintains a state machine within its core, keeping track of transitions between the dynamic phases of the flight vehicle based on the output of its Kalman filter. At a rate of 500Hz, individual systems read and process inertial data from the on-board gyroscopes and accelerometers. At this interval, gyroscopic rates are integrated into a high-resolution attitude quaternion for processing rotational information. A 50Hz data acquisition process operates to read barometric pressure, from which the altitude is determined. It is on this interval that the Kalman filter is stepped, fusing collected positional and inertial data to facilitate state estimation with greater statistical accuracy.

The recovery system is implemented in the target-specific code for application in HIVE's rocket avionics. The apogee state, as defined by the internal state machine, is reached when at least two of three conditions are evaluated to be true: velocity relative to ground is negative, pressure is increasing (altitude is decreasing), and off-launch-axis tilt is greater than 90 degrees. Once apogee is determined, a critical task is notified, and a message is delivered on the flight-system CAN bus with indication of the event that occurred (flight phase has reached apogee) in addition to a voting candidate identifier. The arbiter, operating a minimal application for greater assurance of reliability, is responsible for aggregating all voting messages passed on the bus. Upon assessing a majority vote, the arbiter activates recovery energetics, ensuring both arming and activation signals are asserted, while monitoring continuity to confirm deployment. Through its application of redundancy in multiple firmware and hardware layers, Australis is capable of safely realising deployment events for recovery systems.

RESULTS, CONCLUSIONS AND FUTURE WORK

A system of this complexity requires extensive testing and verification prior to its deployment as a stand-alone flight computer system. Correct electrical operation of the Australis hardware including all sensors, transceivers, and microcontrollers has been verified through a series of isolated unit tests and typical Printed Circuit Board testing procedures. Bench testing has verified internal and external communications systems and additional pyrotechnics testing has confirmed the operation of the recovery deployment circuit. Australis, at a foundational hardware level, has been confirmed to be operational.

While previous data from other flight computers can be fed into the Australis system to simulate a real flight, more extensive flight testing will verify that the Australis flight computer can be responsible for operating flight critical events. Rocket flights can be emulated by using high-performance multirotor drones and system operation can be confirmed without risking failed recovery of a rocket. These tests have proved to be effective and repeatable, exhibiting the least risk to expensive hardware. More progressive tests allow the Australis hardware to operate as a payload to gather real flight data which is analysed against the same COTS flight computer data. This facilitates apogee detection and recovery event timing verification. Following this, flight tests can extend to allow the Australis hardware to assume primary recovery responsibility with COTS redundancy (As will be the configuration at IREC 2025). Future work must involve greater system reliability testing, before finally allowing the Australis System to operate un-aided on future HIVE rocket systems.

REFERENCES

- ¹Choudhary, J., Balasubramanian, P., Varghese, D.M., Singh, D. P., and Maskell, D., “*Generalized Majority Voter Design Method for N-Modular Redundant Systems Used in Mission- and Safety-Critical Applications*” Computers (Basel), 2019-01, Vol.8 (1), p.10
- ²G. Welch and G. Bishop, “An Introduction to the Kalman Filter,” *SIGGRAPH*, 2001, Course 8 (27599-23175), p 24

Development and Validation of a Novel Framework for 6DoF Flight Simulations of a Sounding Rocket

Team 148 Project Technical Presentation to the 2025 IREC

Lorenzo Ricci, Leonardo Brunetti Paoletti, Simone Cargini, and Massimo Della Monica
Department of Mechanical and Aerospace Engineering, Sapienza University of Rome, Via Eudossiana 18, 00184, Rome, Italy

INTRODUCTION

A high-fidelity six degrees of freedom (6DoF) simulation environment has been developed to support the design, analysis, and verification of sounding rocket flights. Its modular architecture allows the reconfiguration of functional blocks to model specific subsystems or mission phases. Features include support for actuator and sensor models, control strategies, and integration with Hardware-in-the-Loop (HIL) systems. The simulator enables rapid prototyping via C and C++ code generation and a visual programming interface, reducing errors in avionics development. It includes stochastic modeling for Monte Carlo analyses to assess mission risk. Built in MATLAB®/Simulink®, the tool leverages model-based design to aid early validation and iterative development.

MODEL CONFIGURATIONS AND ASSUMPTIONS

At the core of any high-fidelity rocket simulation lies the aerodynamic model. Popular open-source simulators often fall short of modeling specific aerodynamic effects with sufficient accuracy. For instance, weather-cocking predictions can be unreliable due to reliance on empirical or simplified theoretical data, which lack the precision of high-level Computational Fluid Dynamics (CFD) analyses or wind tunnel testing. Moreover, several effects, such as aerodynamically induced roll moments, are often completely omitted. To overcome these limitations, a comprehensive aerodynamic model was developed for full 6DoF simulations¹. The model is based on a validated CFD database, wind tunnel experiments, and flight data. Simplifications, such as symmetry assumptions for specific flight configurations and parameter dependencies on roll angle, are incorporated to reduce computational costs while maintaining accuracy².

Another crucial aspect is environmental modeling, which significantly influences apogee prediction and trajectory shaping. We implemented modular gravity and atmospheric models that can be adjusted in complexity depending on the analysis needs. The gravity model includes a baseline spherical harmonics approach combined with centrifugal force effects. The atmospheric model is based on the International Standard Atmosphere (ISA)³, with corrections applied to variables such as lapse rate and Sutherland's constant based on ambient humidity and temperature. Furthermore, wind modeling is customizable, allowing high-fidelity simulation of wind-sensitive flight regimes. Engine modeling for solid student-researched and developed (SRAD) motors and hybrid propulsion systems is also supported, accounting for non-standard geometries and combustion dynamics.

AVIONICS VALIDATION

One of the key motivations for developing a custom 6DoF simulator was the need to test various subsystems of the rocket, particularly the control system. A closed-loop actuation model was created to simulate the servo mechanism and recreate the actual deployment of airbrakes, including potential failure scenarios due to excessive friction. This required a dynamic simulation environment capable of supporting filtering algorithms and state estimation, which justified the use of a full 6DoF framework.

Initially, a Model-in-the-Loop (MIL) setup was implemented, simulating sensor inputs such as an Inertial Measurement Unit (IMU) and a barometric altimeter. These inputs were processed through a Kalman filter⁴ for state estimation. Later, a HIL system was introduced to debug the onboard avionics software. Thanks to Simulink® code generation capabilities, transitioning from MIL to HIL was straightforward, and the same control algorithms could be deployed directly to embedded hardware, significantly reducing human error.

To further evaluate robustness, stochastic simulations were used in the MIL environment to analyze control system behavior under variable atmospheric conditions, engine performance deviations, aerodynamic coefficient

uncertainties, and incorrect mass or inertia estimates. For the HIL setup, real-time communication with the rocket's hardware was established via serial interface, enabling the exchange of simulated sensor data. This allowed us to validate the execution frequencies of the onboard software components: the IMU operated at 200 Hz, the barometric sensor at 80 Hz, the Kalman filter at 200 Hz, and the control algorithm at 25 Hz.

RESULTS, CONCLUSIONS, AND FOLLOW-ON WORK

In addition to the technical results already discussed, the closed-loop simulations provided fundamental insights into the importance of accurate subsystem modeling and how initial assumptions significantly affect system behavior. Through the development of the 6DoF simulator and the validation of the aerodynamic database, we gained a deeper understanding of sounding rocket dynamics and identified key flight risks. Looking ahead, we aim to streamline testing procedures by minimizing manual code interaction, with the long-term goal of implementing a full digital twin of the vehicle. To support this, we are exploring methods to enhance the accuracy and reduce the generation time of aerodynamic coefficients.

REFERENCES

Books

- 1 Barrows, T. M., and Orr, J. S., Dynamics and Simulation of Flexible Rockets, Academic Press, Cambridge, MA, 2020.
- 2 Zipfel, P. H., Modeling and Simulation of Aerospace Vehicle Dynamics, 2nd ed., AIAA Education Series, American Institute of Aeronautics and Astronautics, Reston, VA, 2007.
- 3 U.S. Standard Atmosphere, U.S. Standard Atmosphere, 1976, U.S. Government Printing Office, Washington, D.C., 1976.
- 4 Becker, A., Kalman Filter from the Ground Up, 1st ed., Self-Published, 2023.

Real-Time Apogee Prediction and Correction in Rocket Flight via Active Drag Modulation

Team 172 Project Technical Presentation to the 2025 IREC

Tianjun Wang and Joseph Hadad

Department of Mechanical and Aerospace Engineering, University of Florida, Gainesville, FL, 32612

INTRODUCTION

One of the primary goals of IREC is to predict the apogee of the rocket to a reasonable margin. Typically, this is done through extensive simulations which account for wind conditions, pitch angle on the rod, and other information that is known before flight. An airbrake system is included on the rocket which allows for active control of the rocket's coefficient of drag, flight path, and subsequent apogee during flight. The intention of the system is to enable the launch vehicle to reach within 100 ft of its target apogee of 10,000 ft. Although simulations regarding the vehicles' flight path are accurate, and can predict apogee within a reasonable margin, there are many uncertainties which they simply cannot account for. These uncertainties lead to a model which fails to predict apogee within our desired margin for accuracy. Actively collecting data during flight allows the rocket to calculate a real-time predicted apogee based on true states that will only be known throughout the flight. Based on this information, the airbrake is used to vary two important vehicle characteristics, exposed cross-sectional area and coefficient of drag, to vary the rocket's flight path so that it reaches its desired apogee.

FLIGHT DYNAMICS

To solve for the predicted apogee of the vehicle in real time, a three degrees-of-freedom dynamics model was used to approximate the vehicle's coasting ascent period between motor cutoff and apogee. For the model, it was assumed that longitudinal drag and weight are the only forces acting on the system, the drag force acts through the center of pressure, and the weight acts through the center of gravity. Lateral forces such as wind are ignored, and the pitch angle is based on data collected from flight simulations due to this. It is also assumed that the angle of attack is zero for the duration of the flight as the longitudinal axis of the rocket will generally be aligned with the flight path.

The pitch angle is solved numerically because the rotational dynamics of the launch vehicle are a result of moments caused by the normal forces experienced by the vehicle during flight, which for simplicity's sake have been assumed to be zero for the dynamics model¹. Without access to a wind tunnel to provide accurate data, we determined it would be unreasonable to develop a normal force model. Computational fluid dynamics simulations were used to approximate lift and drag coefficients for rocket components; however, we were unable to produce consistent values and subsequently abandoned the endeavor. Instead, we used RASAero to develop predictions for the pitch angle dynamics over a range of initial conditions with a Monte Carlo simulation.

To model drag, a high-fidelity coefficient of drag model was created for the vehicle that accounts for individual components of the vehicle. The model is based on the U.S. Air Force DATCOM method, which has been utilized to model aircraft and missiles^{2,3}. The drag model assumes that there is zero angle of attack and only considers parasitic drag, ignoring induced drag. Wave and supersonic drag are also ignored, as the flight is assumed to be subsonic.

The parasitic drag considered in the model consisted of skin friction drag (viscous effects), interference drag (body joints and protuberances), and base drag (boattail aft). All components on the exterior of the launch vehicle were analyzed. The coefficient of drag for each component was dependent on the Mach number, geometry of the component, component location on the launch vehicle, and the roughness factor of its material. Bolt heads, holes, and slots in the airframe were lumped together and considered excrescencies. These are assumed to be distributed evenly on the airframe of the launch vehicle and are a function of the surface area of the launch vehicle. The total coefficient of drag was calculated by summing up the drag from all the components of the launch vehicle.

CONTROL STRATEGY AND IMPLEMENTATION

The controller for the airbrake was designed with three main goals: minimize overshoot, maintain steady-state error $\leq 1\%$ (± 100 ft), and rise time not being a constraint. Since the airbrake system is only capable of reducing

apogee, it is an underactuated system⁴. For the controller, this means that any overshoot is not recoverable, leading to the launch vehicle undershooting the target apogee. Thus, the system must minimize the overshoot. For choosing a control law, the system lends itself to a PI controller. The omission of a derivative term is because a high-powered rocket flight during coasting ascent is effectively a massive overdamped system. Because the system is underactuated, constantly decelerating until apogee, tuning the proportional gain such that there is no overshoot and including an integral term for steady-state error provides the control needed.

The error for the controller comes from target apogee and the predicted apogee from the dynamics model. At every iteration of the onboard software, the dynamics model takes the current state inputs of the launch vehicle, solves the model using an RK4 approximation, and returns a predicted apogee. For each iteration, the current airbrake actuation state is used for the first timestep and then assumed to be fully retracted for the rest of simulation. The error signal is then fed into the controller, which then sends the airbrake mechanism an actuation signal. To test and tune the PI controller, a MATLAB Simulink model was altered to include the closed-loop feedback controller. The expected initial conditions on the day of launch were input into the model, and the target apogee was set to 3,048 m (10,000 ft).

HARDWARE DESIGN AND SIMULATION

The airbrake has a pair of flaps that sit on a pair of in-line slider cranks driven by a cam. A high torque servo motor drives the cam. The cranks are connected to the cam symmetrically, which mechanically ensures that the flaps are always extended symmetrically, applying a symmetrical drag on the rocket.

The slots on the airbrake are chosen for a flap's width of 2.5 in and a thickness of 0.25 in. The stress analysis of the flaps is conducted under the maximum loading condition. This happens if the flaps extend immediately after the motor burnout. The air density and velocity were gathered from OpenRocket simulations, with the wind velocities assumed to be 25 mph. The coefficient of drag was assumed to be 1.5, characteristic of a flat plate in perpendicular flow, and that of a wind load was assumed to be 0.01, characteristic of a flat plate in parallel flow. With flaps made of 6061 T6 aluminum, the factor of safety was found to be 11.8, and maximum tip deflection is 0.1 in.

The minimum required torque of the servo motor is also calculated based on the speed of the flaps, the force of friction, and the load. The required torque was found to be 8.8kg•cm. From the torque, a high-torque servo motor with a stall torque of 17.2 kg•cm was chosen.

ELECTRICAL DESIGN AND INTEGRATION

A Raspberry Pi 4 Model B with 8GB RAM was used as a microprocessor. This processor has sufficient power to run the entire control loop within 10ms. Two Parallax MS5607 measure the altitude from barometric pressure. At target apogee of 10,000 ft, it can achieve an accuracy of 10 ft, which is 0.1% of the altitude. A MTI-3 IMU is used to measure the acceleration and pitch angle. The IMU can measure acceleration with a bias of 40 μ G and pitch angle of 0.5 degree. In addition, a separate linear voltage regulator was used to provide 3.3V power to the sensor to minimize the noise from digital components on the board. The barometer and IMU are critical for control inputs and need to be very accurate.

Four BMP390 barometric pressure sensors are used to measure the pressure gradient on top and the bottom of the flaps. Furthermore, two switching regulators provide power at 5V for digital components and 7.4V for the servo motor. A SRAD PCB integrates all the electronics components together. The PCB is mounted on a fiberglass sheet fastened to forward bulkhead and upper plate.

TEST RESULTS AND CONCLUSION

Relevant test flight data analysis revealed that the airbrake successfully deployed for the first time on our third test flight. Despite unintended oscillations in airbrake actuation, the airbrake was able to effectively reduce the launch vehicle's apogee toward the target altitude, achieving an apogee of 4,677 ft in attempting to lower the expected 4,847 ft apogee to a target 4,577 ft. This significant reduction in target apogee validated the control authority that the airbrake has over flight profile.

Flight data indicated that oscillation in the airbrake actuation closely matched oscillations in z-velocity data (derived from pressure sensor data). It was determined that oscillations in z-velocity were a result of the airbrake physically affecting pressure sensor readings. These oscillating z-velocity readings led to oscillating estimates for apogee by the dynamics model and also the physical airbrake extension. To fix this problem, the airbrake has been properly sealed off from the pressure sensor so that interference is mitigated. Changes have also been made to pressure filter coefficients.

REFERENCES

¹S. Niskanen, "Development of an Open Source model rocket simulation software," Pdf, HELSINKI UNIVERSITY OF TECHNOLOGY, 2009. Accessed: Apr. 27, 2025. [Online]. Available: <https://openrocket.sourceforge.net/thesis.pdf>

²R. D. Finck, "USAF (United States Air Force) Stability and Control DATCOM (Data Compendium)," Mar. 1978. Accessed: Apr. 27, 2025. [Online]. Available: <https://apps.dtic.mil/sti/citations/ADB072483>

³R. A. Braeunig, "Drag Coefficient Prediction." 2020.

⁴Russ Tedrake. Underactuated Robotics: Algorithms for Walking, Running, Swimming, Flying, and Manipulation (Course Notes for MIT 6.832). Downloaded on [Apr. 27, 2025] from <https://underactuated.csail.mit.edu/>

Iteration Upon a Closed-Loop Active Drag System for Precision Apogee Control of a Sounding Rocket

Team 190 Project Technical Presentation to the 2025 IREC

Dawsyn Schraiber and Ben Anderson
Virginia Polytechnic Institute and State University, Blacksburg, VA, 24061

INTRODUCTION

Past years' vehicle projects have felt the effects that uncertain weather conditions and manufacturing inconsistencies in COTS solid rocket motors have on the accuracy of predictions with respect to the vehicle's final apogee. In turn, a negative feedback-based control system was developed in the form of the Active Drag System (ADS). The ADS guided the launch vehicle's design toward the overshoot of the desired target apogee to allow for a dynamic reduction of this value via the introduction of additional drag. This system is the third iteration of a previous design with a focus on improving the overall drag control and electronics' performance.

MECHANICAL AND ELECTRICAL DESIGN

The ADS consists of four rack-and-pinion driven carbon-fiber flaps (i.e. air brakes). The four flaps each have a surface area of 26 square inches and have a maximum movable range of 30 degrees from their resting position flush against the rocket body. The rack and pinion are made of lightweight aluminum and attached to the flaps with an aluminum track and ball. The rack and pinion of all four flaps are driven by a central gear attached to a servo motor. This design makes it so all four flaps will all move the same degrees when controlled. The mechanism is held in a 3-D printed housing printed out of ABS plastic. Each flap is secured to the top of four 3-D-printed columns with bolts and lock nuts, which act as the flap's hinge. The entire assembly weighs about 3 lbs and is only 7 in tall. The ADS is located about 3 feet from the aft of the rocket between the rocket motor and the drogue parachute bay.

A custom PCB flight computer was developed featuring a Raspberry Pi RP2040 as the primary, onboard microcontroller. The flight computer holds a TE Connectivity MS5607 barometric altimeter, an Analog Devices ADXL375 accelerometer, a TDK-InvenSense IIM-42653 inertial measurement unit (IMU), and a Memsic MMC5983MA magnetometer as its primary sensors. A 2S4P 7.4 V Lithium-Ion battery is regulated with a 5 V, 8A TPS51386RJNR buck voltage regulator and further a 3.3 V LD39200PU33R low dropout voltage regulator to provide power for the microcontroller and sensors. The flight computer controls the Sincecam 50 kg High Torque servo via Pulse-Width Modulation where different pulse-widths determine the deployment percentage with a forward-voltage diode to prevent current flowing back into the RP2040. In order to reduce power consumption prior to the launch event, an opto-isolated circuit was developed to enable the Raspberry Pi Pico's control over the servo's power supply without additional risk of undesired sink current. This was accomplished via an optocoupler driving an N-Channel Power MOSFET such that the servo's power supply circuit is only explicitly enabled by the RP2040 which occurs at the exit of the launch rail in software.

STABILITY AND CONTROL POLICY

The team utilizes ANSYS Fluent CFD software to simulate pressure distributions of the launch vehicle ascending flight stages. CFD analysis determined that the center of pressure (CP) moves forward when the ADS is fully deployed. However, since the ADS does not deploy until after motor burnout, the forward shift of the center of gravity (CG) results in a stability caliber >1.5 at all stages of ADS deployment. Similarly, the same CFD software was also utilized to simulate the coefficient of drag at finite stages of the ADS' deployment cycle for a lookup table to be used when developing the control scheme.

This year saw the development of a more thoroughly custom flight computer with more sensors than ever, so a real-time operating system (RTOS), i.e. FreeRTOS, was utilized to better manage CPU time and sensor sampling tasks (500 Hz). A Kalman Filter was utilized to fuse altitude data from the onboard barometric altimeter as the measurement and the vertical acceleration as the control (obtained from the onboard IMU's linear acceleration and euler angles vector(s)). The estimated state error covariances were obtained from the manufacturer's datasheets. This provided a stable set of altitude, vertical velocity, and vertical acceleration values for use in the deployment percentage calculations. Additionally, the RP2040's second core was utilized to parallelize data logging to the external flash chip at a rate of 100 Hz. The finite state machine for determining the rocket's flight stage was written

to depend upon a combination of internal hardware timer and altimeter driver interrupts. These implementation details allow for nigh-instantaneous reactions to flight stage events.

For the deployment scheme, this year's design reiterated previous years' establishment of a relationship between an optimal flight path with a 10,000' AGL apogee and the corresponding 'best' coefficient of drag. Two, one-dimensional models were utilized. One considers a constant air density (model A) and the other applies the standard atmosphere condition, as defined by the U.S. Standard Atmosphere (model B)[1]. Both models are used for apogee determination and iteratively estimating the optimal drag coefficient when provided a current altitude (e.g. valid values range from 500' to 10000') and vertical velocity (e.g. valid values range from 0 m/s and 280 m/s). Subsequently, a polynomial fit model was developed to correlate the optimal drag coefficient and the current, calculated drag coefficient to result in a corresponding deployment percentage. This process aims to minimize the number of onboard computations given the operation of the Active Drag System on an underpowered microcontroller. Given a deployment percentage and that the rocket is within the coast phase, the RP2040 is then able to output an appropriate pulse-width modulated signal for the high-torque servo to deploy the air brakes.

RESULTS, CONCLUSIONS, AND FOLLOW-ON WORK

The finite state machine for deployment during the coast phase was validated with video supported with logged data at a test launch on April 27th. We were also able to validate the state machine's functionality operating on only the altimeter via a variable pressure chamber to simulate higher altitudes from the ground. These results give confidence in the ADS' ability to deploy properly for the 2025 Spaceport America Cup. In the future, we hope to move more of the drag coefficient calculations onboard for more accurate real-time results.

REFERENCES

¹Lew, T., Lyck, F. and Müller, G., "Chance-Constrained Optimal Altitude Control of a Rocket" *8th European Conference for Aeronautics and Aerospace Sciences (EUCASS)* [online journal], URL: <https://www.eucass.eu/doi/EUCASS2019-0388.pdf> [cited 6 May 2024].

Design and Manufacturing of Forged Carbon Fiber Flaps

Team 203 Project Technical Presentation to the 2025 IREC

Natalie J. Vu¹

Case Western Reserve University, Cleveland, OH, 44106, United States

Hania A. Abdelhafez²

Case Western Reserve University, Cleveland, OH, 44106, United States

INTRODUCTION

This year was the first time Case Rocket Team attempted to manufacture airbrake flaps for an IREC competition rocket, so the team had to develop a reliable, cost-effective in-house manufacturing process. Ideas considered were CNC machining the flaps or casting the part in metal, but it was decided that using forged carbon fiber would result in a cheaper part that was optimized for minimal weight with respect to strength. The forged carbon process utilizes scrap carbon fiber pieces to which the team already has access from previous projects, limiting the need to purchase additional materials. As for process feasibility, manufacturing a high-strength mold allows for the process to be easily repeatable, since the same mold can be used for all necessary parts. Forged carbon parts are also isotropic¹, making it a good material choice for airbrake flaps as they experience stresses in multiple directions. In order to ensure a repeatable, low-cost, and teachable process, a relatively simple mold surface preparation and release method needed to be developed.

MOLD DESIGN & PREPARATION

The first requirement of the forged carbon project was designing a mold. The flap was initially designed in CAD. The flap geometry was then used to design a five-part compression mold, also in CAD. The mold was composed of a base, cap/plug, and three middle pieces to be bolted together for casting. These design choices were made in order to prevent the final part from getting stuck in the mold due to possible mold-release failure, avoiding destructive part removal and conserving material. Additionally, all pieces included a chamfer to fit a screwdriver for easier removal. The cap piece featured two conical extrusions to mark and pre-center drill the holes eventually needed for flap attachment. The base and three middle pieces formed a female mold part while the cap was the male part. The mold was printed at 45% gyroid infill using PETG in order to maximize strength of the mold and prevent failure while under compression in the curing stage. Following printing, all relevant surfaces on all mold parts were sanded to 2000 grit to remove layer lines and provide a smooth surface finish on the final casted part. The next element of the process design was selecting a mold sealant and release method.

MOLD RELEASE TESTING

For the sake of maintaining a low project cost, two sealant and release pairs were tested. The first was Partall Paste #2 and Universal Mold Release (UMR), while the second was Loctite's Frekote FMS Sealant (FMS) and 770-NC AERO Release (770-NC) system. In order to compare the methods, two molds were printed and sanded as identically as possible. First, four coats of Partall Paste #2 were applied and one coat of UMR was sprayed, both according to their respective package instructions, with the Partall Paste/UMR system being a longer, more involved process than the Frekote system. The mold treated with the Partall Paste/UMR system was loaded first, according to a method combining traditional hand lay-up techniques with compression molding. Laminating epoxy and half-inch chopped carbon fiber tow were distributed across the entirety of the female part of the mold. The male part was brushed with a small amount of epoxy and evenly covered by one layer of the processed carbon fiber by hand (enough to fully absorb the epoxy below it but no more). The total amounts of epoxy and carbon fiber were determined by the volume of the flap, with a goal ratio of 60% carbon fiber to 40% epoxy. The mold was then closed and placed under compression in a table vice with two pieces of wood to evenly distribute the compression force. After being compressed over the course of 5-10 minutes, the mold was left to cure for 24 hours. The use of laminating epoxy eliminated the need for heating throughout the process as typically seen in professional

1. Composites R&D Lead, Mechanical and Aerospace Engineering, 10900 Euclid Ave, Cleveland, OH, 44106
2. Vice President, Biomedical Engineering, 10900 Euclid Ave, Cleveland, OH, 44106

compression molding techniques². The same process was subsequently followed for the mold treated with the FMS and 770-NC.

PART REMOVAL & FINAL APPLICATIONS

Once the parts were cured, the mold was opened using screwdrivers and mallets. For the mold treated with Partall Paste and UMR, removal was much easier than anticipated. The chamfers on all parts proved successful as they were easily separated by a screwdriver. The flap was removed from the final element of the mold via a mallet. There were a few visible voids on the top and bottom of the part, but, since no epoxy or fibers were found stuck to the mold itself, it is likely that these voids were due to loading error since it was the first part made with this method. The FMS/770-NC mold also came apart without needing to break any of its elements, but there was a significant amount of fiber and epoxy stuck to its surfaces. These results indicated that the FMS/770-NC system was not a reliable method for sealing the mold and preparing it for part release. As such, Partall Paste #2 and Universal Mold Release were utilized in the production of all four airbrake flaps. While the initial mold was mechanically fit for reuse, measurement of the final part found dimensional errors due to the draft angles of the mold cavity; thus, a third and final mold was printed, sanded, and treated with Partall Paste and UMR.

The same mold loading method was employed in the manufacturing of all four final flaps, with additional care to evenly distribute the epoxy laterally to prevent additional voids and dry spots. This was successful for three out of four flaps. It was found that the voids on the fourth flap were easily filled and treated by applying a layer of laminating epoxy. Between part castings, any excess epoxy was cleaned off of the mold and two to three coats of Partall Paste and one coat of UMR were reapplied. The same mold was able to be used for the entirety of manufacturing, with only one part needing reprinted after damage from a screwdriver. This part was the smallest and thinnest, making it the most likely to crack but easiest to reprint and sand.

POST PROCESSING

Upon removal from the molds, parts were processed to address voids, dimensional accuracy and surface finish. Initial inspection of the parts revealed excess flashing at the edges, which was manually removed using files. This and all subsequent sanding was done outdoors while wearing a respirator to mitigate particulate inhalation. A spray bottle of water was used to minimize airborne dust during this process.

After flashing removal, any remaining excess material was smoothed using 120 grit sandpaper. This stage also served to eliminate minor surface imperfections introduced during curing. Particular attention was paid to regions where part dimensions were more important, such as the slot interfaces for airbrake flap attachment—these areas often required resanding to bring them within specification. Maintaining tight tolerances proved difficult as material removal due to sanding and common tolerancing problems with 3D printing were somewhat unpredictable.

Following sanding, holes were located via marking and drilling a pilot hole. This allowed for a better centering and an easier milling process. Then they were drilled to the complete width and depth at the predetermined locations using a mill. In some cases, dry spots in the carbon layup led to small surface voids. These voids were filled using a thin, brushed-on layer of laminating epoxy. A fine bristle brush was used to distribute the epoxy evenly and minimize pooling. Once cured, all previously drilled holes were redrilled.

The final surface finishing process involved successive sanding steps, progressing from 120 grit to as fine as 2000 grit, to achieve a smooth and uniform finish. This level of surface refinement was necessary to meet aerodynamic performance requirements.

RESULTS, CONCLUSIONS, AND FOLLOW-UP WORK

The development of this compression mold process for forged carbon parts has enabled the team to produce experimental non-geometric flaps with greater material efficiency and consistency. The method supports rapid iteration and reuse, with minimal reliance on specialty tooling, since everything used was available in-house. Planned improvements include optimizing the ratio of excess epoxy, fiber fineness, reducing cavities to minimize required post-processing, and validating part performance and surface finish under simulated flight conditions. Mold release tests also optimized the mold removal process, which indicated Partall Paste/UMR to be the preferred method. This process provides a solid foundation for expanding forged carbon manufacturing across other structural and aerodynamic components within the team's vehicle architecture, and provides other teams a simple and consistent method for producing specialty components.

REFERENCES

1. Andrew Y. Chen, Sebastian Baehr, Austin Turner, Zilan Zhang, Grace X. Gu, Carbon-fiber reinforced polymer composites: A comparison of manufacturing methods on mechanical properties, *International Journal of Lightweight Materials and Manufacture*, Volume 4, Issue 4, 2021, Pages 468-479, ISSN 2588-8404, <https://doi.org/10.1016/j.ijlmm.2021.04.001> [cited April 23, 2025].
2. Jamir, M., Majid, M, and Khasri, A., *Sustainable Composites for Aerospace Applications*, Woodhead Publishing, Cambridge, England, UK, 2018, pp. 160-163.

Development of a Non-pyrotechnic Recovery System for High-powered Rockets

Team 204 Project Technical Presentation to the 2025 IREC

Nischal Chirkute* and Pawarit Maneeratanadamkirng†

Chulalongkorn University High Altitude Research, Faculty of Engineering, Bangkok, 10330, Thailand

INTRODUCTION

The safe and reliable recovery of high-powered rockets is a critical component of modern rocketry, ensuring both the protection of valuable onboard systems and the ability to reuse key components. Traditionally, recovery systems have relied heavily on pyrotechnic devices such as black powder charges to deploy parachutes. While effective, these systems introduce significant safety risks, regulatory constraints, and logistical challenges related to storage, transport, and handling of explosive materials.

The Chulalongkorn University High Altitude Research Club is dedicated to upgrading our launch vehicle's recovery system by exploring safer and more efficient alternatives to conventional pyrotechnic methods. In line with this objective, this report presents the development of a non-pyrotechnic recovery system specifically designed for high-powered rocketry applications. The goal is to engineer a system that maintains deployment reliability while eliminating the hazards associated with explosive actuation.

The proposed system eliminates pyrotechnic methods and instead employs a fully mechanical and pneumatic design to achieve safe and controlled deployment. Central to this system is a spring-loaded puncture mechanism that activates a CO₂ cartridge, rapidly pressurizing the recovery bay and triggering structural separation of the rocket stages. In place of traditional explosive charges, this controlled release of gas provides a safe and reusable method for initiating recovery events.

DESIGNING THE SPRING INTEGRATED DEPLOYMENT

The Spring Integrated Deployment (SID) is a CO₂-based ejection system designed to pressurize the airframe for stage separation. It serves as an improved iteration of the CURSR-III SRAD ejection device. The mechanism uses a spring-loaded pin to puncture the CO₂ cartridge, chosen for its simplicity, availability, and mechanical reliability.

Two springs are used: a main spring to drive the puncture pin and a rebound spring to prevent the pin from obstructing the gas release. To ensure sufficient force, the spring was sized to exceed the CO₂ seal's ultimate tensile strength (18 MPa), applying a safety factor of 2. This resulted in a minimum required spring constant of 1380 N/m over a 26 mm displacement.

The SID includes a servo-actuated safety and release system. The puncture pin, designed similarly to a syringe plunger, is pulled upward to compress the spring. A mechanical key locks it in position, secured by a secondary safety key. Both keys are sequentially released by a servo arm—rotating counterclockwise to release the safety key, then clockwise to fire the main key. The MG996R servo motor, rated at 11 kg·cm torque, safely exceeds the 4.47 kg·cm required for actuation.

DESIGNING THE ROTOR LEASH

The Rotor Leash, developed as a non-pyrotechnic equivalent to a tender descender, was engineered to provide a reliable, reusable, and mechanically simple means of controlling parachute deployment in high-powered rockets. Its core function is to safely deploy the main parachute at a precise moment during flight, at which point it can be

* Head of Recovery, Chulalongkorn University High Altitude Research Club, 6638107721@student.chula.ac.th

† Recovery Engineer, Chulalongkorn University High Altitude Research Club, 6638119221@student.chula.ac.th

disengaged through servo actuation. Unlike conventional pyrotechnic devices, the Rotor Leash leverages a fully mechanical rotary locking mechanism to achieve safe and repeatable deployment.

The design consists of four primary components: the top case, bottom case, and two interlocking rotor arms (left and right), all machined from Aluminum 7075. This material was selected due to its high strength-to-weight ratio and excellent stiffness, making it suitable for aerospace applications where mass is a critical constraint. Each rotor arm is mounted on a central pivot and features a hooked geometry that allows it to engage securely with its counterpart. When rotated into the locked position, the arms create a mechanical interlock that can withstand the tensile loads experienced during descent.

Actuation of the system is performed by a high-torque TD-8135MG servo motor, chosen to exceed the minimum torque required to disengage the latch under maximum expected tension. To ensure reliable performance, torque calculations were based on a worst-case load scenario, assuming the full dry mass of the rocket during terminal descent. The estimated resistive torque of 2.95 N·m was well within the servo's capacity of over 4 N·m, providing a significant safety margin.

Attention was also given to mechanical alignment and vibration damping. Bushings were integrated into the pivot points to ensure precise rotation and prevent misalignment under flight loads. The use of standard M8 bolts, lock washers, and nuts enables ease of maintenance and reusability, allowing the system to be disassembled and reassembled between launches without specialized tools.

To validate the design, finite element analysis (FEA) was conducted to simulate stress distributions and displacements during operation. The results confirmed that the component remained within acceptable safety margins under all tested loading conditions. The combination of material selection, mechanical redundancy, and simulation-based verification resulted in a lightweight, durable, and safe mechanism capable of reliably facilitating parachute deployment in a non-pyrotechnic recovery system.

TEST AND MANUFACTURING

The manufacturing process for both the Rotor Leash and the SID emphasized precision, repeatability, and structural integrity under flight conditions. All primary components were machined using 3-axis CNC milling. After fabrication, the components underwent a series of bench tests to validate their mechanical function and fit. Initial assembly tests focused on verifying the alignment and motion of the rotor arms, as well as the effectiveness of the servo-actuated unlocking sequence. The TD-8135MG servo motor was tested under incremental loads to confirm that it could reliably overcome the estimated unlocking torque requirement of 2.95 N·m. Additionally, repeated actuation tests were conducted to evaluate mechanical wear, servo repeatability, and potential misalignments due to vibration or play in the joints.

To simulate flight conditions, the assembled Rotor Leash was subjected to a series of static load tests, approximating the tension forces encountered during terminal descent. The device was mounted to a vertical rig, and calibrated weights equivalent to the rocket's dry mass were suspended from the latch point. These tests confirmed that the interlocking rotor arms could securely retain the load without slippage or deformation. Following static testing, the servo was activated to trigger deployment, and the disengagement motion was evaluated for responsiveness, smoothness, and mechanical integrity.

RESULTS, CONCLUSIONS, AND FOLLOW-ON WORK

In conclusion, both the Rotor Leash system demonstrates strong potential as a safe and reusable non-pyrotechnic method for the recovery system in high-powered rocketry. The system successfully eliminates combustion-related risks, reduces thermal stress on surrounding components, and improves lifetime of the recovery components.

REFERENCES

- [1] AZoM, “Lead (Pb) - Properties, Applications,” AZoM, Jul 12 2013. <https://www.azom.com/article.aspx?ArticleID=9100>
- [2] “MG996R Servo Motor,” Components101. <https://components101.com/motors/mg996r-servo-motor-datasheet>
- [3] Aslan, N. K., “Design and Performance Analysis of a CO₂Ejection System for High-Powered Rockets,” *American Institute of Aeronautics and Astronautics, Inc.*, 2023. <https://doi.org/10.2514/6.2023-71967>

Applied Machine Learning-Based Additive Manufacturing and Topology Optimization for Structural Aerospace Components

Team 208 Project Technical Presentation to the 2025 IREC

Austin Prevette and Neil Sanipara
Mississippi State University, Mississippi State, MS, 39762

INTRODUCTION

Additive manufacturing (AM) enables fabrication of structures that are impractical through traditional manufacturing methods. These additive designs can utilize self-supporting Topology Optimization (TopOpt) to drastically increase specific yield and specific 1st mode. The resulting mass-efficient designs reduce operational stress by up to ten times compared to their subtractive counterparts, as verified by a NASA study¹. These AM+TopOpt designs are meta-optimized using Kriging-based engineering surrogate modeling techniques; introducing machine learning (ML) into the design process reduces the design time from weeks to hours. The resulting novel components are rigorously tested against various criteria—including yield failure testing, random vibration testing, and real-world flight tests. The compressive strength and vibrational response of these parts were verified at Mississippi State University's Center for Advanced Vehicular Systems (CAVS) and NASA Marshall Space Flight Center (MSFC), respectively. Based on the results of the validation tests, the AM components exceed specified testing criteria and will thus be flown on the Helios rocket for IREC 2025, pioneering the technology in applied student aerospace engineering. Furthermore, the reduced mass of these parts allows for extra redundancy in Helios' recovery system.

MACHINE-ASSISTED DESIGN AND AFT TRUSS

Using the same core regressing Kriging ML algorithm—Izanagi—from an internal fin optimization software, various parameterized parts were optimized. Izanagi creates a sample plan for parameterized designs, solves the relevant finite element or fluid dynamics, then curve fits the data based on scaled gaussian curves. The prime example, the Aft Truss, is a compressive member that transfers engine thrust into the rocket body. The Aft truss has two interface surfaces: one surface containing 6 (qty) ¼-20 through-hole fasteners for attachment to the rocket thrust ring and the other containing 12 (qty) 8-32 threaded holes for the aeropack retainer. It must also fit within a 2.75 in x 5 in (height and radius) cylindrical volume. This part must be low mass, structurally efficient, and have no modes below 1 kHz. AlSi10Mg was chosen as the build material to fit those requirements. Because of AlSi10Mg's superior compatibility with metal powder bed fusion (PBF) technologies, high cost efficiency, and high specific vibrational strength, it was the best choice for the trusses. To compare the power of Izanagi, a human-made Aft Truss was designed in tandem as a base truth. Then, various computational and experimental tests were performed to quantify the benefit of machine-assisted parts.

AFT TRUSS COMPUTATIONAL TESTING

To determine how the parts would perform in real life, Ansys was used to make a digital twin of both Aft Trusses. First, the masses of either part were determined before the parts were manufactured. The machine-assisted truss is 281 g, and the engineered part is 340g. Then, a modal test was performed on both trusses. The first mode for the machine assisted truss is 1634 Hz, and the engineered is 1886.1 Hz. While higher in raw magnitude, the specific 1st mode of the machine-assisted truss is better.

Next, the yielding of the material was predicted using Ansys nonlinear static structural. A compressive force is applied in the axial direction, with the bottom face fixed as mirrors the real-life integration. Then, the force is increased at each new timestep, and the maximum Von-Mises stress recorded. Once this stress surpasses 270 MPa, or the yield strength of the material, the simulation was terminated. From those tests, the machine-assisted truss withstood 52 kN, and the engineered truss withstood 43 kN. Thus, the machine-assisted truss was nearly 50% better in specific yield strength.

AFT TRUSS EXPERIMENTAL TESTING

To further verify the computational results, both Aft Trusses endured non-destructive fixed vibrational testing at NASA and destructive testing at CAVS.

A test fixture for the trusses was produced in-house, designed such that it could also be used for multiple test articles. Results were obtained from a tri-axial response accelerometer, which was placed on the top ring for each truss. Additionally, two tri-axial accelerometers were placed on the fixture itself as controls. For the sine sweeps, plots of acceleration versus frequency were recorded. For the random tests, plots of ASD versus frequency were recorded. Overall, no significant modes are observed on any of the vibration testing plots, indicating that either truss can easily withstand the vibration sustained in the launch environment. Since both trusses had no notable modes below 2 kHz, and the topology optimized truss is lighter, it is evident that this truss has superior mass-specific performance.

Furthermore, compressive failure testing at CAVS established the factors of safety of the Aft Trusses. Both trusses were placed on square steel test plates and loaded in the test machine loading space in two separate runs. Axial loading was applied on the structures at a constant strain rate until significant failure occurred. The data from the test suggests the machine-assisted truss started to yield at 40 MPa applied stress, while the engineered truss started to yield at 35 MPa. While the machine-assisted truss failed before the engineered truss in the test, it only did so due to post-processed drilled holes. With a static F.S. of more than 28, both trusses were certified for flight.

PAYLOAD TOP SHELF AND FLIGHT COMPUTER MOUNT

While not directly involved in a trade study, the payload top shelf and flight computer mount were designed with the IZANAGI algorithm and printed out of AlSi10Mg and PETG, respectively. Thus, the entire rocket has components from the boat tail to the nosecone that were created with machine-assisted design. Because the payload needs to be a calibrated mass, the top shelf was designed to hit a precise mass target, as well as increase the vibrational stiffness of the payload. The optimization algorithm was able to create a part that is the best blend of vibrational stiffness and mass. Furthermore, the flight computer mount in the avionics coupler midbay was designed with IZANAGI. By distributing mass efficiently, the new flight computer mount weighs 2x less and fits in a form factor 2x smaller than heritage flight computer mounts.

CONCLUSION

Technological advances as much as fractions of a percent are praised in the aerospace domain. ML models and machine-assisted structures prove that even with hundreds of years of engineering, order of magnitude improvements in design are still possible. In unpublished results, the block 3 manufacturing process of the Aft Truss has produced a truss that beats the engineered truss by 6x the operational stress, corroborating the NASA study's results. Thus, topology optimized ML structures provide the structural breakthrough that could lead to another revolution in manufacturing. As the price of additively manufactured components continues to decrease, and applications of space-based manufacturing are explored, the authors suggest further exploration of the domain. Future research in this direction should focus on validating the block 3 and 4 parts made with IZANAGI, as well as exploring methods to quantify the fatigue life of parts made with additively manufactured materials. Additionally, more research into the applications of these parts into high performance aerospace and automotive will provide the confidence in part life and durability over time.

REFERENCES

[1] NASA Goddard, "*Generative Design and Digital Manufacturing: Using AI and Robots to Build Lightweight Instruments,*" NASA Goddard Space Flight Center, <https://ntrs.nasa.gov/api/citations/20220012523/downloads/McClelland-Generative%20Design%20SPIE%202022.pdf>

FEM Validation of an Additively Manufactured Flange: Onyx vs. PEEK

Team 212 Project Technical Presentation to the 2025 IREC

Emir Topakci, Andrea Manzo, Edoardo Viglietti, Marco Donà, Andrea Pantano
*Dipartimento di Ingegneria Meccanica e Aerospaziale, Politecnico di Torino, Corso Duca degli Abruzzi 24, Torino,
10129, Italy*

Introduction

Recent advancements in both Additive Manufacturing (AM) technologies and the materials used have significantly improved the mechanical performance of 3D-printed parts, expanding their applicability in structural applications. PoliTo Rocket Team (PRT), the student rocketry team of Politecnico di Torino, leverages these advantages by incorporating AM into many of its structural components. The team has access to advanced equipment, including Markforged Mark Two, and Apium P220, as well as high-performance materials such as Markforged Onyx, and Ensinger polyether ether ketone (PEEK) CF 30. Despite the promising outcomes, accurately simulating the mechanical behavior of 3D-printed parts remains challenging due to the anisotropic and heterogeneous nature of the printing process. Nevertheless, by adopting widely accepted modeling approaches from the literature, it is possible to achieve simulations that closely approximate real-world behavior.

Mechanical Properties

The mechanical properties of materials are strictly dependent on the nature of Filament Fused Fabrication (FFF). The main factors that have an effect are the raster angle, raster orientation, raster type, shell of the parts, infill percentage and the adhesion between the layers. Even though much research has been done on these parameters, since there are many independent variables, the properties are mostly condition dependent [1].

The effect of both raster and infill will be neglected in the plane since the specimens and flange have been printed with equilateral triangular infill. However, the property will be decreased in the built direction because of possible layer adhesion problems that are commonly faced in the FFF processes [2]. Furthermore, the effect of the shell will be considered, and homogenization of the properties of infill and shell will be applied [3].

Theoretical properties for Onyx [4] and PEEK [5] have been provided by the respective manufacturer. Mechanical properties of materials have been decreased in the building direction of 22% [6]. Also, the infill (37%) was considered, and the properties had been scaled down following the infill percentage [7].

Tensile Test Specimen

A series of tensile tests have been conducted following the ASTM D638 specimen's standard to evaluate the mechanical properties of materials. Furthermore, these results have been used to validate the finite element model of the specimen and eventually the flange.

First, a High Fidelity Model (HFM) was done on the specimen, modelling separately the shell and the infill with constraining one grip section and applying force on the other grip section. Subsequently, stress is calculated along longitudinal direction dividing force by area. Eventually, stress is divided by the strain in the longitudinal direction which has been given by the simulation to get the Elastic Modulus (E).

Thereafter, Rule of Mixture (ROM) was used for the Low Fidelity Model (LFM) to average the mechanical properties that use the volume fractions for shell and infill. After averaging each orthotropic property with these, a new simulation has been run with one set of elements instead of separate elements for shell and infill. Subsequently, the same theoretical approach has been used to calculate the E. Furthermore, for the PEEK specimen, since it was 100% infill, the simulation has been done with only one set of elements.

Motor Retention System

The Motor Retention System (MORES) is one of the most critical flanges of the rocket, playing a pivotal role transferring thrust from the solid motor to the aerostructure.

PRT's Vittorio Emanuele II Mark 2 rocket MORES is designed to withstand a peak thrust of 7551 N generated by an Aerotech O5500X solid rocket motor. The part was designed and optimized using topology optimization software *nTop, Release 4.1, nTop Inc.*

A detailed material comparison was necessary to select the most suitable option between Onyx and PEEK. To validate the design, a compression test was carried out using an Instron 68SC-5 testing machine. This test simulated the motor's thrust, applying a reduced peak force of 5000 N due to maximum capacity of the machine. The flange was fastened with screws to a section of the rocket's body tube, which was fixed at the base of the testing machine, while the upper bracket applied the force on the loaded surfaces.

A linear static analysis of the flange was done using the LFM with orthotropic materials via *Altair Hypermesh, Version 2025, Altair Engineering Inc.* To ensure reliable results, the boundary conditions were defined by applying fixed constraint in the mesh nodes located at the interfaces of the threaded inserts, which connect the flange to eight M5 screws. Given the complexity of the geometry, a 3D tetrahedral mesh was selected with a low average element size to preserve the geometry details.

Non-Flange Displacements - $u_0(F)$

The displacement $u_0(F)$ is defined as a function of the applied force F , representing the total non-flange displacement measured during testing. This displacement comprises the combined elastic deformations occurring in the test tube, the testing machine, the screws, and the threaded inserts. Subsequently, a system of three linear equations is introduced with three different unknowns: elastic displacement results in Onyx and PEEK flanges ($u_{el,onyx}(F)$, $u_{el,PEEK}(F)$) and $u_0(F)$. Thereafter, this system is solved for $F=5000N$ to calculate $u_0(F=5000N)$. Following this, u_0 is linearly scaled to the force values. Finally, $u_0(F)$ is added to corresponding results from LFM for flanges.

Comparison Of Results

The results from the compression test were compared to the LFM which has been predicted by the specimen test results. A force-compressive displacement curve was plotted using *MatLab, Version R2024b, The MathWorks Inc.*, and the most significant parameter extracted was the slope of the interpolated first-order polynomial. *Figure 1* illustrates that, after adding the u_0 value to the LFM results, the behavior of the flanges can be predicted.

The LFM result curve matches experimental results with acceptable accuracy, the major difference noticed is that this curve is a first-order polynomial, due to the assumption of linearity of the numerical calculations, while the experimental results are not. In addition, the Onyx prediction curve is more precise than that of PEEK, suggesting that the LFM has higher accuracy with the former material.

Conclusions, And Follow-On Work

The comparative study of Onyx and PEEK materials, supported by both experimental testing and finite element modeling, provided valuable insight into material selection and structural behavior.

The compression test confirmed that LFM rather correctly reflects the stiffness of the flange using both Onyx and PEEK, with results aligned with the experimental data. Moreover, the model underestimates the stiffness of the flange for both materials, which is desirable from an engineering perspective.

An important variable to consider is the deformation of the testing machine. Under load, the machine and its subcomponents tend to deform due to non-ideal material behavior. Furthermore, deformations which happen in the tube, screws and threaded inserts should also be taken into consideration. For that reason, the u_0 numerical calculations were introduced. This integration gave satisfactory results, despite the strong hypothesis of system linearity.

Overall, the combination of testing, numerical modeling, and topology optimization validates the flange design under realistic loading conditions and confirms its use in flight. PEEK emerges as the preferred material, due to its superior mechanical response under load.

Future work may include dynamic analysis, improved orthotropic material characterization, and non-linear system, for non-flange displacement prediction to further enhance model fidelity and performance predictability.

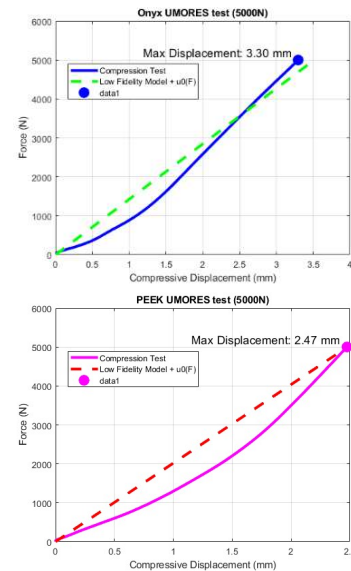


Figure 1

References

- [1] "(PDF) Optimising the FDM Additive Manufacturing Process to Achieve Maximum Tensile Strength: A State-of-the-Art Review," *ResearchGate*. <https://doi.org/10.1108/RPJ-07-2018-0183>
- [2] Ma, C., Faust, J., and Roy-Mayhew, J. D., "Drivers of Mechanical Performance Variance in 3D-Printed Fused Filament Fabrication Parts: An Onyx FR Case Study," *Polymer Composites*, Vol. 42, No. 9, 2021, pp. 4786–4794. <https://doi.org/10.1002/pc.26187>
- [3] Kalova, M., Rusnakova, S., Krzikalla, D., Mesicek, J., Tomasek, R., Podeprelova, A., Rosicky, J., and Pagac, M., "3D Printed Hollow Off-Axis Profiles Based on Carbon Fiber-Reinforced Polymers: Mechanical Testing and Finite Element Method Analysis," *Polymers*, Vol. 13, No. 17, 2021, p. 2949. <https://doi.org/10.3390/polym13172949>
- [4] Markforged, "Onyx - Composite 3D Printing Material." Retrieved 1 May 2025. <https://markforged.com/materials/plastics/onyx>
- [5] Ensinger, "PEEK - TECAPEEK CF30 Black | Ensinger." Retrieved 1 May 2025. <https://www.ensingerplastics.com/it-it/semilavorati/prodotti-semilavorati/peek-tecapeek-cf30-black>
- [6] Essentium, "Essentium HTN-CF25 Technical Data Sheet." <https://nexa3d.com/wp-content/uploads/2024/07/TDS-Essentium-HTN-CF25.pdf>
- [7] Jiang, C., and Zhao, G.-F., "A Preliminary Study of 3D Printing on Rock Mechanics," *Rock Mechanics and Rock Engineering*, Vol. 48, No. 3, 2015, pp. 1041–1050. <https://doi.org/10.1007/s00603-014-0612-y>

Lightweight Neural Compression of Precomputed Flight Configurations for Model-Based Apogee Control Systems in High-Powered Rockets

Team 216 Project Technical Presentation to the 2025 IREC

Patrick Bellamy, Anindo Minifie, Nicholas C. Morrison, Kelvin Liu, Jamie Vu, and Phan M. Q. Nguyen
ARES Rocketry Team, Melbourne, VIC, 3010

INTRODUCTION

Achieving precision apogee control in high-powered rocketry presents a unique challenge due to the complex interplay of uncertain environmental conditions, limited onboard computational resources, and the irreversible nature of aerodynamic braking. In the context of the 30,000 ft Commercial Off-The-Shelf (COTS) category of the International Rocket Engineering Competition (IREC), ARES has developed a lightweight, neural-network-assisted apogee control system designed to consistently achieve target altitude across diverse flight scenarios. Rather than relying solely on traditional physics-based controllers or infeasibly large precomputed lookup tables, our approach compresses petabytes of high-fidelity trajectory data into a compact neural network that runs in real-time onboard the flight computer. This model-based control architecture allows for rapid, accurate predictions of final apogee from current flight states and actuator positions, enabling the system to incrementally converge on the target altitude during ascent.

NEURAL COMPRESSION OF PRECOMPUTED FLIGHT TRAJECTORIES

Traditional control strategies rely on physics-based models to compute actuator commands. While well-established and effective in many aerospace applications, these models can be brittle and difficult to tune in small rockets, where uncertain conditions and limited computational power and storage make real-time evaluation of high-fidelity dynamic models computationally impractical.

An alternative approach is to precompute optimal airbrake extensions across a wide range of possible flight states using a high-fidelity ground-based simulation framework. This allows in-flight control decisions to be made through simple state lookup. Unfortunately, even a lookup table spanning just a limited subset of flight state variables, such as altitude, velocity, and orientation, would require over 30 petabytes of storage—more than double the data required to store all videos ever uploaded to YouTube. Any realistic control system would require extending the state space to account for environmental variation, more state variables, and system-specific behaviors, rapidly compounding the storage requirements beyond what could feasibly be stored onboard.³ Non-neural compression methods are thus unsuitable for this application, as they require all data to be generated, stored, and compressed in one pass, whilst introducing large approximation errors.

To address this, we trained a fully connected neural network on the high-accuracy outputs of simulations generated using RocketPy, a widely adopted, open-source trajectory simulation framework for high-powered rocketry. RocketPy incorporates validated atmospheric models and has been extensively tested and peer-reviewed by the rocketry community, making it a reliable foundation for control development. Its open-source nature also allowed us to modify the codebase to suit our specific vehicle dynamics and flight scenarios.¹

The neural network itself serves as a compact function approximator; an algorithm inspired by the structure of the human brain, capable of learning complex nonlinear relationships between inputs (e.g., flight state and airbrake extension) and outputs (predicted apogee). Once trained, it replaces the need for an infeasibly large lookup table by providing near-instantaneous predictions of apogee with minimal computational overhead. Our model achieves a worst-case prediction error of less than 5 m relative to the underlying RocketPy simulation and occupies less than one-fortieth the memory footprint of the PDF file containing this abstract.²

NEURAL NETWORK ENABLED CONTROL SYSTEM IMPLEMENTATION

This predictive model is embedded into our real-time apogee control system. During flight, the neural network is used to estimate the final apogee based on the current flight state and airbrake extension. The control system uses this prediction to determine the necessary airbrake extension that will converge the projected apogee toward 30,000 ft. Importantly, the system follows a convergence strategy, adjusting extension incrementally to approach the target from above. This avoids undershooting, a terminal failure mode due to the irreversibility of energy loss.

The control architecture is composed of two interlinked components: a high-frequency inner-loop airbrake controller and a lower-frequency outer-loop guidance controller. Both rely on a single neural network model (NN1), trained to predict final apogee from current flight conditions and extension. The airbrake controller, running at 10Hz, uses a binary search over NN1 to determine the extension, which results in the desired apogee. This replaces traditional inverse dynamics or PID-style control with a lightweight, model-based solution that is both more accurate and significantly faster,⁴ as querying NN1 requires only a few matrix multiplications.

An additional benefit of using a neural network is its ability to generalize across a wide range of flight conditions. NN1 is trained on billions of simulated flights across varying environmental models, including turbulence, wind shear, and temperature profiles. As a result, it learns a robust mapping that implicitly accounts for model uncertainty and environmental variance. This robustness enables the controller to operate effectively under diverse and unpredictable conditions, with performance that would be impossible to replicate with traditional control systems.

The outer-loop guidance controller, operating at 1 Hz, determines the optimal target apogee for each control cycle. Rather than always aiming for 30,000 ft directly, it selects a target within an apogee envelope defined by two curves that converge toward 30,000 ft with altitude. This envelope acts as a dynamic bound on the predicted apogee, wide at low altitude to allow aggressive correction, and narrowing at higher altitudes to enforce smaller, precise adjustments. This ensures smooth, monotonic convergence without overshooting. The controller evaluates the control authority error, which is the difference between the expected and realized effect on predicted apogee due to deployment, and maps this error through a sigmoid function. The resulting bounded value is used to interpolate the next target apogee within the envelope. This approach ensures stable convergence, avoids oscillation, and progressively degrades control effort as authority diminishes.

TESTING AND VALIDATION

To validate this approach, we developed a Monte Carlo testing suite that varies off-the-rod velocities and angles within expected tolerances for our motor and simulates thousands of flight profiles across a range of atmospheric conditions and models.^{5,6} Both the neural network and control logic are iteratively refined to maximize the proportion of Monte Carlo flights that reach within ± 30 ft of the target apogee. This framework enables rapid iteration and provides detailed statistical reporting on controller performance.

In preparation for competition, a test article of the current rocket was flown on an Aerotech M2500, alongside last year's test vehicle on an Aerotech M2050, both equipped with identical apogee control surfaces. Across three flights, successive iterations of the control model were evaluated, each demonstrating the expected system response.

The first test flight used last year's rocket and an early prototype of the controller. Due to a launch rail issue, the rocket lifted off at a suboptimal trajectory, ultimately undershooting the desired apogee. As intended, the airbrakes did not deploy, correctly identifying that the vehicle lacked the energy to reach the target altitude.

During the second test flight, the rocket reached an apogee of 3010 m, overshooting the target by 10 m. Although a more advanced version of the control model was used, insufficient atmospheric modeling near apogee led to an overestimation of available control authority, which caused the system to undercorrect.

Our next test launch is scheduled for the 4th of May, during which we will fly the latest iteration of our model. This version incorporates improved state variable polarity in its training data, more accurate, Monte Carlo-validated atmospheric modeling, and integrates the finalized control architecture as previously described.

RESULTS, CONCLUSIONS, AND FOLLOW-ON WORK

The system presented offers a novel, lightweight, and highly effective solution to model-based apogee control, leveraging neural compression to overcome the prohibitive storage and computational demands of traditional physics-based methods. In flight, the controller executes thousands of predictions per second on embedded hardware, operating well within real-time constraints. By approximating precomputed high-fidelity flight configurations, our validated architecture enables robust, precision altitude control. As hardware capabilities continue to improve, this approach scales naturally to more demanding applications⁷, including precision landing, responsive launch systems, and other time-critical aerospace control tasks.

REFERENCES

This work was enabled by the use of open-source development tools and frameworks including Python, TensorFlow, NumPy, and Git for version control. These tools, while not directly referenced in the text, played an essential role in building, training, and validating the proposed system.

¹Ribeiro, D., “RocketPy: A Trajectory Simulation Tool for High-Powered Rocketry,” *Journal of Open Source Software*, Vol. 5, No. 52, 2020, Paper 2413.

²Goodfellow, I., Bengio, Y., and Courville, A., *Deep Learning*, MIT Press, Cambridge, MA, 2016, Chaps. 6–9.

³Narayanan, A., and Shmatikov, V., “Privacy and Security: Myths and Fallacies of ‘Personally Identifiable Information’,” *Communications of the ACM*, Vol. 53, No. 6, 2010, pp. 24–26.

⁴Levine, S., Finn, C., Darrell, T., and Abbeel, P., “End-to-End Training of Deep Visuomotor Policies,” *Journal of Machine Learning Research*, Vol. 17, No. 39, 2016, pp. 1–40.

⁵Rubinstein, R. Y., and Kroese, D. P., *Simulation and the Monte Carlo Method*, 2nd ed., Wiley, New York, 2007, Chap. 3.

⁶Anderson, J. D., *Fundamentals of Aerodynamics*, 5th ed., McGraw–Hill, New York, 2011, Chaps. 4, 7.

⁷Farabet, C., Poulet, C., Han, J. Y., and LeCun, Y., “CNP: An FPGA-Based Processor for Convolutional Networks,” *Field-Programmable Logic and Applications*, FPL, IEEE, 2009, pp. 32–37.

⁸OpenAI, Akkaya, I., Andrychowicz, M., et al., “Solving Rubik’s Cube with a Robot Hand,” arXiv preprint, arXiv:1910.07113 [online], URL: <https://arxiv.org/abs/1910.07113> [cited 1 May 2025].

⁹Pagano, A., “Model Predictive Control of Rocket Flight for Precision Apogee Targeting,” *AIAA Guidance, Navigation, and Control Conference*, CP-2003-5585, Vol. 1, AIAA, Washington, DC, 2003, pp. 231–239.

¹⁰Abadi, M., et al., “TensorFlow: Large-Scale Machine Learning on Heterogeneous Distributed Systems,” arXiv preprint, arXiv:1603.04467 [online], URL: <https://arxiv.org/abs/1603.04467> [cited 1 May 2025].

¹¹Harris, C. R., et al., “Array Programming with NumPy,” *Nature*, Vol. 585, 2020, pp. 357–362.

¹²Rossum, G. van, and Drake, F. L., *Python 3 Reference Manual*, Python Software Foundation, Scotts Valley, CA, 2009.

¹³Chacon, S., and Straub, B., *Pro Git*, 2nd ed., Apress, New York, 2014.

Development and Testing of an Innovative Multi-Layer Plate Deployment (MLPD) Mechanism for Precision Altitude Control

Team 218 Project Technical Presentation to the 2025 IREC

Sumit Chanda, Santiago Prendiz, Victor Bonilla, Esteban Ruiz, David Moreno and Pedro M. Ferreira da Silva
Sun City Summit Rocket Team, The University of Texas at El Paso, El Paso, Texas, 79902

INTRODUCTION

Airbrakes are aerodynamic control surfaces used to increase drag and decelerate a vehicle during flight, commonly employed in aerospace systems for altitude and velocity control. Their effectiveness in modulating ascent profiles makes them critical components in rocket recovery and precision landing systems. Traditional airbrake designs can be often categorized into sliding and hinged-plate airbrakes¹. Sliding airbrakes function by extending a plate laterally from the rocket body to create additional drag; however, they suffer from limited deployment area due to constraints on maximum size by rocket body tube, leading to reduced braking effectiveness. Hinged airbrakes, on the other hand, rotate outward from the rocket body, providing a larger deployed surface area and improved drag generation. Despite this advantage, hinged designs require a greater portion of the rocket's outer structure to be cut away to accommodate the full range of fin motion, introducing challenges to the structural integrity of the airframe. Additionally, hinged mechanisms are mechanically more complex and susceptible to aerodynamic loading issues at high speeds. Both designs face trade-offs between achievable braking force, structural robustness, and mechanical reliability, limiting their performance under demanding flight conditions. To address these challenges, a novel Multi-Layer Plate Deployment (MLPD) airbrake system has been developed. This design incorporates a sandwich-like structure, allowing multiple plates to be stacked within a compact volume and sequentially deployed using variable motion controlled by a single actuator shaft. The MLPD airbrake increases the effective braking surface area without requiring large cutouts in the rocket body, preserving structural integrity. By solving the area limitations of sliding airbrakes and minimizing the time required for deployment, the MLPD system enhances flight reliability and improves aerodynamic braking performance for precision altitude control.

DEVELOPMENT OF THE MULTI-LAYER MECHANISM

As part of the Special Project at Sun City Summit Rocket Team, the Multi-Layer Plate Deployment (MLPD) mechanism was developed as an innovative solution to overcome the surface area limitations of traditional sliding based airbrake designs while maintaining a compact stowed profile. Conventional sliding airbrakes, commonly driven by electrically actuated mechanisms such as Hoberman linkages, origami flasher patterns, rack-and-pinion systems, cam-follower assemblies, and gear trains, rely on passive motor rotation to extend and retract the drag surfaces^{2,3}. Among these, the Hoberman mechanism is widely favored for its simplicity and reliable actuation, where the motor's rotational motion is translated into linkage movement to deploy and retract the drag plates⁴. However, in these systems, the maximum effective drag area is constrained by the rocket's outer diameter, as the plates must remain nested inside the body when stowed, oriented perpendicular to the rocket's length. To address this limitation, the MLPD mechanism was conceptualized and developed by integrating a stacked, multi-layer plate approach. Two alternative designs were initially explored using CAD modeling to increase the drag area within the same body constraints. The first approach involved foldable plates that would unfold upon deployment, while the second employed a sandwich-like configuration with multiple layers of plates stacked vertically. Each layer could extend to a different radius from the rocket's body, enabled by a multi-tiered coupling mechanism. The final MLPD design utilizes a custom actuator shaft with varying coupling diameters, where each layer of coupling connects to linkages of different lengths. This configuration allows the upper plates to achieve greater displacement from the body than the lower ones during deployment. To ensure structural stability of the extended upper plates, each includes a slide-in rod that passes through a guiding slot in the lower plate, providing mechanical support once fully deployed. The MLPD system is powered by a single motor that actuates all layers simultaneously through the shared actuator shaft. This design preserves the

simplicity of a single-actuator mechanism while achieving a larger effective braking surface. Furthermore, the layered deployment eliminates the need for complex unfolding sequences required by the foldable plate concept, improving reliability and reducing potential points of failure during flight. By enabling a greater deployed surface area within the same cross-sectional envelope, the MLPD airbrake enhances drag performance and reduces deployment time, both critical for improving the precision and reliability of altitude control in high-speed flight applications.

CFD ANALYSIS

A computational fluid dynamics (CFD) analysis was performed to evaluate the aerodynamic forces acting on the Multi-Layer Plate Deployment (MLPD) airbrake system during flight. The simulation domain was designed to replicate the plenum and exterior flow conditions experienced by the rocket. Boundary conditions were established based on the recorded flight parameters of the test rocket, including inlet velocities, operating altitude, and atmospheric pressure. The inlet flow velocities were defined according to the rocket's expected velocity profile during ascent, ensuring accurate representation of the aerodynamic environment. The airbrake geometry was imported directly from the CAD model into the CFD software to preserve dimensional fidelity. Turbulence models were selected to capture the high-Reynolds number flow regime surrounding the deployed plates. Both steady-state and transient simulations were conducted to observe the drag forces across various deployment angles. These CFD predictions were later used as input for the structural evaluation and design validation of the airbrake system.

MLPD FABRICATION

The Multi-Layer Plate Deployment (MLPD) airbrake system was fabricated primarily using additive manufacturing techniques to produce the intricate components and linkage mechanisms with high geometric precision. The main body, deployment plates, couplers, and linkages were 3D printed using a high-strength Carbon PLA filament to achieve a lightweight yet durable structure suitable for flight testing. Static load tests were performed on the fabricated airbrake system by applying forces equivalent to the drag forces predicted by the CFD analysis on each deployed plate. These tests verified the mechanical integrity and ensured that the linkage system could withstand the expected aerodynamic loads during operation. Integrating the airbrake into the rocket body presented a unique challenge due to the complex socket shapes required for plate deployment. Cutting these sockets directly into the cylindrical cardboard rocket body was difficult and risked compromising structural stability. To overcome this, the rocket fuselage was split longitudinally, and a custom 3D-printed coupling sleeve was designed and fabricated to bridge the two halves while providing precision openings for the airbrake plates to extend. This solution maintained the rocket's structural integrity while allowing smooth and reliable deployment of the airbrake system during flight.

RESULTS, CONCLUSION, AND FUTURE WORK

The Multi-Layer Plate Deployment (MLPD) airbrake system underwent extensive testing through both computational and experimental evaluations. The initial single-plate airbrake design provided a surface area of 3708 mm², while the multi-layered mechanism increased the total deployable area by approximately 75%, offering a significant improvement in braking surface without increasing the system's footprint. CFD analysis at the rocket's maximum flight velocity of 267 m/s showed that the original single-plate design generated 338 N of drag, while the MLPD system produced 471 N, increasing the drag coefficient from 0.69 to 0.96. Two flight tests were conducted to validate the system. In the first test, the airbrake did not deploy due to a sensor malfunction, though the rocket was successfully recovered. In the second flight, the airbrake deployed as intended, effectively reducing the apogee according to predictions, with no observed stability issues or structural failures. The prototype successfully demonstrated the deployment and retraction of multiple plates at varying radii during both ground tests and in-flight operation, confirming the mechanical robustness and functional viability of the design.

This work has demonstrated the feasibility of a multi-layered airbrake system capable of increasing drag efficiency using a single actuator-driven mechanism, contributing to improved altitude control for high-powered rockets. The successful deployment and performance of the MLPD system highlight its potential not only as an innovative solution for compact, efficient airbrake designs, but also in other aerospace applications where it demands for greater surface areas folded/stored inside a compact volume like space-craft antennas/heat shields etc.

Building on these results, the team is currently working on scaling the MLPD airbrake for integration into the IREC competition rocket for higher altitude and more demanding flight conditions.

REFERENCES

- ¹ Nesrin, S., "Preliminary Design Review," University of California, Davis – SpaceED Rockets Team, 2013.
- ² Lee, H. and Alam, P., "The Design of Carbon Fibre Composite Origami Airbrakes for Endeavour's Darwin I Rocket," *Journal of Composites Science*, Vol. 5, No. 6, 2021, p. 147.
- ³ Zavorotny, A. and Lin, A., "Team 55 Project Technical Report for the 2018 Spaceport America Cup," University of Ottawa Student Team of Aeronautics and Rocketry, 2018, pp. 17-19.
- ⁴ Lin, B., and Hou, J., "Design of Bistable Deployable Scissor Structures Consisting of Translational Units Based on Flat Retraction Logic," *CAADRIA 2023*, Volume 2, pp. 541-550.

Nose Cone Manufacturing Using Vacuum Lamination Technique

Team 302 Project Technical Report for the 2025 IREC

UFABC Rocket Design - 2025 IREC Team
Universidade Federal do ABC, Santo André, São Paulo, 09210-580, Brazil

INTRODUCTION

The design and manufacturing of the fuselage for the Aracê Project involved a comprehensive process that combined theoretical analysis, computational modeling, and practical manufacturing techniques. The primary objective was to develop a fuselage that would ensure the structural integrity of the rocket while optimizing aerodynamic performance under flight conditions. This process was characterized by iterative testing, refinement, and validation, which were crucial in achieving the desired design specifications.

DESIGN PROCESS

The initial phase of the project focused on determining the geometric parameters of the fuselage. To facilitate the design process, the team employed simulation tools such as OpenRocket and Ansys CFD to model the flight characteristics and analyze the aerodynamic behavior of various fuselage shapes. The simulations provided valuable insights into the flow dynamics, pressure distribution, and potential shockwave formation around the fuselage. Several geometric configurations were tested, and based on the analysis, the Haack Series design with a length of 35 cm was selected as the most optimal configuration for the Aracê rocket.

The choice of this geometry was informed by its superior aerodynamic efficiency, as well as its compatibility with the overall structural requirements of the rocket. Once the geometric parameters were finalized, the focus shifted to the manufacturing methodology. After careful consideration of various fabrication techniques, vacuum lamination was selected due to its ability to provide the necessary mechanical strength and surface finish while remaining cost-effective. The fuselage would be fabricated using fiberglass and epoxy resin, employing a mold (gel coat + fiberglass) and a 3D-printed counter-mold to achieve the desired shape and surface finish.

ANALYSIS AND TESTING

With the design and manufacturing method established, the team moved to the practical stage of fabricating the molds. The first task was the creation of the counter-mold, which was designed using CAD software. The counter-mold was printed in multiple sections, taking into account the size limitations of the 3D printer. This required careful planning to ensure that each section of the mold aligned correctly during assembly. Once the counter-mold was printed, the team applied a resin treatment to achieve the necessary surface finish. The process involved multiple stages of resin application followed by curing and sanding to obtain a smooth and uniform surface.

Simultaneously, the team prepared the primary mold. The process began with the application of a Gel Coat and resin mixture to the counter-mold, followed by the layering of fiberglass. This procedure ensured that the mold would have sufficient structural strength to withstand the forces experienced during the rocket's flight. A key aspect of this stage was the precise timing required to prevent the Gel Coat from hardening prematurely, which would hinder proper adhesion. After curing, the mold was carefully de-molded, and any excess material was trimmed and sanded to meet the specifications.

The lamination process began with the preparation of the fiberglass fabric, which was cut according to the dimensions derived from the CAD model. The fabric was then impregnated with a resin-catalyst mixture, ensuring that each layer of fiberglass was properly saturated. The lamination proceeded in a methodical

manner, with each layer of fiberglass being placed in pairs, ensuring that the laminate structure would have the necessary mechanical properties. The application of resin was done using a variety of tools, including spatulas and rulers, to ensure even distribution of the material.

The mold assembly was then completed, and the final layers of the fiberglass laminate were placed within the vacuum bag. A breather layer was added to absorb excess resin, and the vacuum bag was sealed to create the necessary pressure for the lamination process. The vacuum pump was operated for approximately 12 hours to ensure complete resin curing and to remove any air pockets that could compromise the structural integrity of the fuselage. This phase of the manufacturing process was critical to achieving the desired mechanical properties and aerodynamic characteristics.

RESULTS, CONCLUSIONS, AND FOLLOW-ON WORK

The completed fuselage was subjected to a series of evaluations, which confirmed that the manufacturing process had been successful in producing a high-quality, robust structure. The vacuum lamination method provided the necessary strength and surface finish, while the 3D-printed counter-mold facilitated precise fabrication. However, the process was not without challenges. One significant issue was the size limitations of the 3D printer, which required the counter-mold to be divided into multiple sections. To address this, alignment flanges and pre-marked screw holes were added to ensure proper assembly.

In terms of future work, the team identified several areas for improvement. One potential modification is the use of ABS material for direct printing of the mold, which would eliminate the need for Gel Coat application. Additionally, the use of liquid rubber as an alternative to Gel Coat was considered, as it could simplify the manufacturing process. These alternative methods will be further explored in subsequent phases of the project to enhance both the efficiency and effectiveness of the fuselage production process.

The successful completion of the fuselage manufacturing process for the Aracê Project represents a significant milestone in the development of the rocket. The design, analysis, and testing procedures have laid a solid foundation for future advancements in rocket construction, providing valuable insights that will inform future projects. The lessons learned during this phase will contribute to the continued refinement of the team's manufacturing techniques and aerodynamic designs.

REFERENCES

1. OpenRocket Team, OpenRocket Technical Documentation, OpenRocket Project, 2013. URL: <https://openrocket.info/documentation.html> [Accessed 22 December 2024].
2. OpenRocket Team, OpenRocket User Guide, OpenRocket Project, n.d. URL: https://openrocket.readthedocs.io/en/latest/user_guide/basic_rocket_design.html [Accessed 6 January 2025].
3. Ansys, Inc., Aerodynamics of a Rocket Using Ansys Fluent, Innovation Space, n.d. URL: <https://innovationspace.ansys.com/product/aerodynamics-of-a-rocket-using-ansys-fluent/> [Accessed 6 January 2025].
4. Ansys, Inc., How Simulation Changed the Rocket Engine Design Cycle, Ansys Blog, 2019. URL: <https://www.ansys.com/blog/simulation-changed-rocket-engine-design-cycle> [Accessed 4 February 2025].
5. SAE International, Composite Materials Handbook – Volume 1: Polymer Matrix Composites Guidelines for Characterization of Structural Materials, CMH-17, 2012.
6. Strong, A. B., Fundamentals of Composites Manufacturing: Materials, Methods and Applications, 2nd ed., Society of Manufacturing Engineers (SME), 2008.
7. Metamorph Software, OpenRocket Case Study | OpenMETA, OpenMETA Project, n.d. URL: <https://openmeta.metamorphsoftware.com/case-studies/rocket/> [Accessed 4 February 2025].

Oblique Shockwave and Pressure Gradient Analysis of a Von Karmen Nose Cone

Team 306 Project Technical Presentation to the 2025 IREC

Weston Suhadolnik

Kent State University College of Aeronautics and Engineering, Kent, OH, 44266, United States

Golden Flashes Rocketry Team

Kent State University College of Aeronautics and Engineering, Kent, OH, 44266, United States

Introduction

Accurate and high-precision pressure measurements are important in the transonic to low-supersonic flight regime due to the increased dynamic pressure variations. These variations have both aerodynamic performance and stability impacts, especially involved in high-speed rocketry applications. This study focused on characterizing the pressure gradient and shockwave properties over the 5:1 Von Karmen nose cone chosen for this year's competition rocket—a critical region for aerodynamic analysis—using both theoretical and computational methods. The overarching goal of the system being designing a model capable of reading and corroborating known simulation and theoretical data to bolster a comprehensive and technical understanding of shockwaves.

Importance

The need for this analysis stems from predicted velocity data obtained through simulations and previous test flights, indicating the rocket will pass through high-speed regimes where pressure spikes are prominent due to Mach pressure divergence. Test flight data showcased that Andromeda entered the transonic and low-supersonic regime, the area where this system is potent. This application lends itself outside of the high-power rocketry field into a vast array of technical areas. Understanding how these spikes work and why they form are especially relevant for aerospace and defense applications, where oblique shockwaves can lead to large wake formation and flow separation, resulting in air-intake inefficiencies.

Theoretical Considerations

Theoretical pressure values were calculated using the standard barometric pressure equation, incorporating altitude variations and physical constants. This data was inputted into both MATLAB simulations to create a background of comparison for the data that the system will be creating. This allowed for the development of a pressure gradient along the nose cone surface with the goal being corroborated with flight results. Demonstration results showed elevated pressure at the tip—where initial oblique shockwaves form—and a gradual decrease toward the base, consistent with well-documented normal shock behavior.

Simulation and Validation

To validate these predictions, CFD simulations were conducted in OpenFOAM under low-supersonic, compressible flow conditions using environmental parameters from Midland, Texas. These included a freestream velocity of 350 m/s and a temperature of 320 K, corresponding to aerodynamic and ambient characteristics obtained from other team subsystems. Two models were analyzed: one with a solid nose cone and another featuring drilled holes for surface routed pressure sensors. Simulation results demonstrated a peak pressure of 170–180 kPa at the

nose tip, tapering to approximately 110 kPa at the base. These results closely aligned with MATLAB analytical calculations, with a percentage difference of ~9%. This further establishes a baseline of comparison, meaning that flight data should fall within the expected range of these simulations.

Conclusion

To briefly conclude, the pressure altitude and gradient measuring system will employ onboard pressure transducers during flight to collect real-time data for validating these theoretical and simulation results. This will support post-flight reconstruction of shockwave behavior and help evaluate the aerodynamic effects of sensor ports, especially in complex geometrical regions like a nose cone. Although this is a largely experimental study, the goal is to enhance understanding of oblique shockwave dynamics in supersonic flight to inform future nose cone and air-intake designs for high-speed aerospace vehicles.

REFERENCES

¹G. Sarwar, P. Kumar, "Supersonic flow investigation of drag reducing spike on blunted nose body", *Institution of Mechanical Engineers*. [online journal] Vol. 237, No. 9, Paper 1, URL: <https://journals.sagepub.com/doi/10.1177/09544100221140942?icid=int.sj-full-text.similar-articles.1> [Cited 26 April 2025]

²J. Anderson, "Fundamentals of Aerodynamics," *Oblique Shocks and Expansion Waves*, 6th ed., Vol 1, McGraw Hill Education, New York, 2017, pp. 414-616

Laboratory Analysis of Solid Rocket Motor Nozzle Slag

Team 312 Technical Presentation to the 2025 IREC

Olivia Lira¹ and Kyle Markel²
Texas Tech University, Lubbock, Texas, 79409

INTRODUCTION

Ammonium Perchlorate Composite Propellant (APCP) is one of the most common types of propellant used in solid rocket motors (SRMs), primarily consisting of ammonium perchlorate oxidizer, hydroxyl-terminated polybutadiene binder, and a curative. Additives to modify the burn rate or make the propellant easier to mix or pour are also commonly used. In addition, aluminum powder is often included, serving as an additional fuel, as it can significantly increase a propellant's efficiency.

The use of aluminum in APCPs causes the phenomenon of slag buildup on the motor's nozzle, which occurs as the molten aluminum and aluminum oxide exits the nozzle and solidifies, creating a mass of metal in and on the throat and diverging section of the nozzle. During the full-scale static fire of the Space Raiders motor for IREC 2025, a significant amount of slag was observed. The team proposed to investigate the slag in more detail, both to build the team's knowledge surrounding the behavior of aluminum in the team's specific SRMs, but also to contribute to the scientific and amateur rocketry communities in learning about these types of motors. To that end, the team pursued laboratory characterization techniques to learn about the composition and morphology of the observed motor slag. Specifically, the team undertook X-Ray Diffraction (XRD) and Scanning Electron Microscopy (SEM)/Energy Dispersive X-Ray Spectroscopy (EDS) testing to characterize samples of the motor slag.

X-RAY DIFFRACTION

X-ray diffraction is a nondestructive analytical method used to discover physical properties of solid materials. It is widely accepted as one of the best methods to obtain chemical composition, phase composition, and crystal structure. XRD works by projecting the X-rays onto the sample at a defined angle and recording the intensity of the diffraction angles. Using Bragg's Law and the diffraction angles, the interplanar spacing of the crystal structure can be calculated, revealing information about the sample's crystal structure. Additionally, the Scherrer formula relates the width of diffraction peaks to crystalline size, providing insight into what the chemical composition of the solid could be.

The team chose to use XRD because of its nondestructive quality and the extensive data that could be extracted. For this study the team chose to scan two samples of the slag because of its two distinct sections: the diverging and throat sections. The diverging section exhibited a dark color and brittle quality, while the throat section was lighter in color and significantly harder. During sample preparation, we broke the nozzle slag to powderize the samples. This revealed a cross section with three layers; a thick white layer sandwiched between two thinner black layers. This observation prompted the team to hypothesize that the white layer was aluminum oxide, a common post burn product of aluminum.

Once the XRD scans concluded, we worked with a specialist to analyze the diffraction graphs and peaks. The graphs revealed peaks that very closely matched the diffraction peaks of aluminum oxide, validating our initial hypothesis. Although both samples had significant amounts of aluminum oxide, a quantitative analysis revealed a 12% increase in aluminum oxide from the throat section to the diverging section. The team concludes this could be due to the temperature variations or supersonic flow through the nozzle. Additionally, a significant amount of carbon was found in both samples.

¹ ollira@ttu.edu

² kylejm220@gmail.com

SCANNING ELECTRON MICROSCOPY (SEM) AND ENERGY DISPERSIVE X-RAY SPECTROSCOPY (EDS)

Scanning Electron Microscopy (SEM) allows for visual investigation of the surface of a material. A scanning electron microscope exposes a sample to a beam of electrons, and by measuring the response of the electrons to the material, an image can be formed with very high magnification and resolution. In addition, by pairing SEM with EDS, or Energy Dispersive X-Ray Spectroscopy, the elemental composition of the material can also be determined. EDS works similarly, but analyzes the X-rays produced by the sample when exposed to an electron beam.

For analysis of the motor slag, two different samples were prepared, one from the nozzle throat area and one from near the aft end of the nozzle diverging section. With the knowledge that these two locations are exposed to two very different environments during the course of a motor burn, the team wanted to obtain data that would best enable comparison between the two locations while also obtaining as much information about the material as possible. Therefore, the team elected to primarily focus on obtaining SEM imaging and EDS data for the cross-sections for both of these samples. In preparing these samples, the team noted an interesting cross-section feature. Visually, the cross-section was made up of roughly three layers: two black layers sandwiching a white layer. The team believed that the white layer was a mixture of alumina and pure aluminum and assumed that the black layer was likely carbon-containing. However, in obtaining the SEM and EDS data for these cross-sections, it was discovered that the black layers were in fact also primarily composed of alumina and pure aluminum (though there was a higher carbon concentration in this location). In addition, it can be understood by general visual inspection of the SEM images that the material has a highly pocketed and porous structure.

In addition, the team was also interested in the composition of the slag on the inside of the diverging section of the nozzle, and therefore did SEM imaging and obtained EDS data for the surface of the sample. This confirmed that the surface had a higher carbon content than the central part of the cross-section, which was consistent with visual observations. It is also notable that the morphology of this surface is very different than the morphology observed in the cross-section. On this surface, aluminum tended to form more spherical deposits, carbon tended to build up more heavily, and the copper formed spherical depositions of varying size.

RESULTS, CONCLUSIONS, AND FOLLOW-ON WORK

The results of XRD, SEM, and EDS helped the team develop a qualitative analysis of the slag produced during our full-scale test fire. XRD revealed a high concentration of alumina present throughout the whole sample, but a 12% decrease was observed from the throat to diverging section. The team concludes this gradient could be linked to the fluid dynamics within the nozzle. SEM and EDS validated these conclusions as we found the presence of alumina, aluminum, and carbon in the analyzed samples. The gradient discovered during XRD was also observed in EDS, with a higher concentration of alumina at the diverging section than the throat. Our analysis prompts us to conclude that the gradients and layers of the nozzle slag are tied to the combustion phases that occur during the burn. The thin black layer touching the nozzle formed during ignition, the middle white section built up during the bulk of the burn, and the outermost layer formed during the burn off. Moving forward, the team plans to continue the characterization campaign by using higher precision techniques such as Raman Spectroscopy or higher resolution SEM imaging. Overall, our analysis has deepened our understanding of the combustion process, what conditions influence slag build up, and how we can improve our propellant formulation.

Machwave: a Stochastic Solid Rocket Motor Simulation Software

Team 313 Podium Session to the 2025 IREC

Felipe Bogaerts de Mattos

Universidade Federal de Juiz de Fora, Juiz de Fora, MG, Brazil, 36036-900

INTRODUCTION

One of the primary challenges in solid rocket motor (SRM) design is ensuring safe, reliable operation in the face of tightly coupled, nonlinear parameters. Small deviations in throat diameter, propellant mass or grain geometry, or burn rate coefficients can yield significant variations in thrust and chamber pressure results - or, in the worst case, lead to a catastrophic failure (CATO). This sensitivity accentuates the need for a stochastic simulation environment, which can deal with uncertainties and physical imperfections in a way that most software currently in use by university-level teams does not support.

MACHWAVE

Machwave [1] is an open source Python library that unifies the key stages of SRM design - internal ballistics, thermochemical properties, structural sizing, preliminary trajectory estimation, and uncertainty quantification. It is modular and extensible, meaning it can be coupled with other libraries or existing software. It was initially designed to integrate several different tools into one, in order to reduce imprecision, make design iterations less time consuming, and avoid introducing human error when switching simulation tools.

Some of Machwave's main attributes are:

- Internal ballistics transient simulation solver with finite-difference grain regression;
- Monte Carlo uncertainty propagation over geometry, propellant properties, and operating conditions;
- Structural sizing of retainer mechanisms and pressure vessels;
- Export of thrust, pressure and mass-flow histories for trajectory integration;
- Three-dimensional trajectory simulation.

Machwave's internal ballistics solver employs a 4th-order Runge–Kutta (RK4) integrator based on Hans Seidel's mass balance equation for chamber pressure modeling [2]. Isentropic relations are calculated using Anderson's Modern Compressible Flow [3]. The RK4 integrator provides a key advantage, enabling high-fidelity simulation of transient phenomena, particularly in grain segments with varying geometries.

MONTE CARLO SIMULATIONS

A Monte Carlo simulation is a stochastic method that randomly samples inputs throughout a specific range to run repeated simulations in a loop. The results can then be aggregated and specific metrics can be analysed in terms of probability distributions.

Machwave's Monte Carlo engine lets you quantify the impact of manufacturing and environmental variability in a single, automated workflow. Users can specify any input parameter as a distribution with a range and a distribution type (e.g. Gaussian, uniform, log-normal). Some examples of input parameters are grain dimensions, throat diameter, burn rate coefficients, propellant density, ambient temperature or pressure, casing thickness tolerances, and more.

The main class of this module, *MonteCarloSimulation*, is responsible for generating scenarios and running the simulation in a pre-determined amount of times. Each scenario is generated by a method that searches for *MonteCarloParameters* inside the simulation parameters recursively, and replaces it with a random value within the given range, according to a probability distribution. The scenario is then used to run the internal ballistic simulation. The results are analyzed using histograms.

By combining the Monte Carlo, internal ballistics and trajectory simulations, it is possible to obtain several key parameters to the design of the motor and the rocket. Some of the most important ones are the total impulse, the

maximum operating chamber pressure, the maximum thrust, the apogee, the maximum velocity, and the maximum acceleration. These parameters can be visualized using the built-in methods to generate histograms, generally producing an approximation of a Gaussian curve.

ÍCARO SRM DEVELOPMENT

Machwave's Monte Carlo module was extensively used when developing Ícaro's SRM. The largest source of uncertainty was the grain geometry, since the propellant manufacturing and molding process is manual and thus physical imperfections can be introduced and propagated in the propellant segments.

The impact of those imprecisions in the chamber pressure is perhaps the single most important factor to consider, since overpressurization can provoke a CATO. When simulating the motor Ícaro with a 5 mm tolerance in the length and inner diameter of each grain segment, the simulation resulted in 3.00 MPa with $\sigma = 0.15$ MPa ($\pm 10\%$ at 2σ). This result is acceptable and demonstrates that the combustion process in this motor is relatively stable and unlikely to produce structural failure in the casing, which is designed to handle internal pressures of up to 9MPa.

RESULTS, CONCLUSIONS, AND FOLLOW-ON WORK

Stochastic simulations are essential for characterizing uncertainty in SRM performance and operational outcomes. Using Machwave's Monte Carlo module, teams can quantitatively model these uncertainties to produce robust risk assessments and reliable performance forecasts.

The stochastic simulations performed when designing Ícaro were crucial to gaining a better understanding of the operation outcomes. However, future simulations could include the introduction of more uncertain parameters as inputs, in order to produce a model that is closer to reality.

Future work on Machwave will focus on expanding its propulsion modeling capabilities to new engine types, strengthening its data-analysis toolkit, broadening its suite of simulation classes, enriching user guidance, and optimizing computational efficiency. These planned improvements include:

- Support for hybrid/liquid rocket engines;
- Better tools for analyzing Monte Carlo simulation data;
- Addition of more simulation classes;
- Improving examples and documentation;
- Performance improvements.

REFERENCES

¹Felipe Bogaerts de Mattos, "Machwave: an extensible Python framework for rocket engine simulations", GitHub repository, v0.5.0, <https://github.com/felipebogaertsm/machwave>, accessed day May 1, 2025.

²H. H. Seidel, "Transient Chamber Pressure and Thrust in Solid Rocket Motors," Brown Engineering Co. Inc. Research Labs, Huntsville, AL, Technical Report AD0613962, 1965.

³J. D. Anderson Jr., Modern Compressible Flow: With Historical Perspective, 3rd ed., McGraw-Hill, New York, NY, 2002.

Active Roll Stabilization using Canard Control Surfaces

Team 400 Project Technical Presentation to the 2025 IREC

Connor B. Mitchell and Rohan M. Joshi
Duke AERO Avionics Team, Durham, NC, 27708

Sage A. Cooley, Joseph C. Schwartz, Grant D. Weerts
Duke AERO Executive Team, Durham, NC, 27708

INTRODUCTION

Perseus, Duke AERO's rocket for the 2025 IREC competition, marks the team's first vehicle to incorporate active roll-control, a milestone toward achieving full in-flight control authority. Central to this system is the integration of a tri-canard control surface array, positioned forward of the variable drag airbrake system. The design, selection, and implementation of this canard-based roll-control mechanism showcases innovation under real-world constraints of mass, power, reliability, and regulatory compliance. This abstract details the technical rationale, system architecture, and expected performance of the canard control system, culminating in a robust, fault-tolerant mechanism that prioritizes both control fidelity and mechanical integrity.

PURPOSE AND BACKGROUND

As launch vehicle missions increase in complexity, fine-grain control of flight dynamics becomes essential, not only for trajectory optimization, but also for safety, stability, and recovery. Until now, Duke AERO's rockets have relied solely on passive aerodynamic stabilization from fixed fins on the aft end of the rocket. However, these methods offer limited control authority, particularly in variable wind conditions or during multi-phase ascent profiles. The implementation of active roll-control in *Perseus* reflects a targeted advancement toward full six-degree-of-freedom flight control, beginning with precise roll-rate modulation.

Following the successful implementation of airbrakes, Duke AERO sought to incorporate an additional flight optimization control system. The canard system serves as a foundational capability toward the exploration of increased trajectory control in more advanced vehicles. Roll-axis control was selected as a logical entry point due to its relative mechanical simplicity and ability to minimize tail fin-induced oscillations and off-axis forces during the coast phase. The engineering objective was to develop a robust, regulation-compliant system that will reliably provide real-time roll control without compromising aerodynamic performance or introducing uncontrolled torques.

CONTROL INTERFACE MECHANICAL SELECTION AND DESIGN TRADEOFFS

The initial design phase for active roll stabilization involved a rigorous comparison between a reaction-flywheel system and an aerodynamic surface-based solution. Although the flywheel presented advantages in terms of aerodynamic cleanliness, it introduced critical drawbacks, most notably high mass, significant power consumption, and complex integration within the fuselage. Flywheel modeling would require spinning a 1.8 kg near-hoop structure at over 5100 RPM to achieve the required moment of inertia ($0.0096 \text{ N}\cdot\text{m}^2$), raising concerns about structural mounting under flight-induced oscillations and excessive energy draw. These constraints led the team to abandon the flywheel in favor of a mechanically-simpler and energetically-leaner canard system.

Once the aerodynamic solution was selected, attention turned to the mechanical architecture required to synchronize the three canards in accordance with IREC Rule DTEG 7.1, which mandates that no controllable pitching moment be introduced by active control systems. Two designs were considered: a belt-driven actuation system and a hybrid model combining a four-bar linkage with a central beveled gear. While the belt system avoided backlash, it posed challenges in maintaining consistent tension and alignment across all blades, risking desynchronization and subsequent instability. The hybrid system, although introducing some mechanical play, proved more robust to alignment issues and allowed for carefully bounded backlash that was ultimately leveraged to ensure passive return-to-neutral behavior under power loss conditions.

INTEGRATION, DYNAMIC RESPONSE, AND FAULT TOLERANCE

The chosen canard system utilizes trapezoidally profiled NACA 0006 aerofoils, selected for their low drag and minimal vortex shedding at supersonic speeds. Pivot points for the canards are placed slightly forward of their aerodynamic centers of pressure, ensuring that in the event of mechanical or power failure, the fins naturally return to a near-vertical, neutral orientation. This design choice compromises increased control torque for significant gains in fault tolerance and dynamic stability.

Structurally, the canard housing is integrated into the upper airbrake assembly with precision-machined running fits and secured via multiple bolt patterns, ensuring structural continuity and load transfer. Finite Element Analysis (FEA) confirmed a minimum factor of safety (FoS) of 20 under expected loading conditions, while the airbrake-coupled section maintained a FoS of 2. The mechanical linkage system includes a 3:1 beveled gear reduction, which introduces a precisely calibrated range of backlash. This backlash serves a dual purpose: it dampens control signal noise and allows the system to self-center within a narrow deadband, mitigating vibrational excitation and oscillation growth. Dynamic stability is further enhanced by statically and aerodynamically balancing each blade around its pivot axis. Blade mass distribution is such that the center of mass aligns closely with the pivot point, while the center of pressure remains slightly aft, creating a passively damped oscillatory system. This ensures that any free-play movement remains within the backlash limit, promoting rapid re-stabilization.

AERODYNAMIC SIMULATIONS (CFD)

The aerodynamic challenges accompanying the canard project were twofold: determining the optimal placement of the canards along the rocket body and choosing a safe pivot point for the shaft connection on the canard blade. The first challenge was solved by sweeping through a range of canard placements: 61, 59, 57, and 55 inches from the aft end of the rocket. These locations are all relatively close to the center of pressure of the rocket, ensuring minimal pitching moment. Coupled with the canard placements, various angles of attack (1° , 3° , and 5°) were considered, forming a total of 18 different test cases.

The results from these cases revealed that the 61-inch placement was optimal because it minimizes the rocket's overall aerodynamic drag and possesses the greatest variance in drag force throughout its cants. Pressure contours produced in all test cases expressed the same conclusion: the canards had little impact on the downstream flow over the fins. This insight confirmed that the canards' primary aerodynamic effect is localized around their own immediate geometry, with a minute impact on the aerodynamic stability of the aft section of the rocket. Next, a complementary study was conducted to determine a safe pivot point for the canards. The goal was to ensure that the center of pressure of the canards would remain below the pivot point throughout ascent, thereby maintaining stability in the event of a control system failure.

The results of the CFD showed that the center of pressure at maximum velocity is 39 mm (1.535 in) from the bottom of the canard fin. Consequently, the canards were designed with a pivot point positioned just above this point, ensuring that the canard's neutral position would not be a detriment to the stability of the rocket.

CONTROL SYSTEM

The canards implement a Proportional-Integral (PI) Control Loop. Using RocketPy, the rocket's flight path and kinematics could be simulated in the context of varied environmental factors. In simulation, several control systems were tested to determine the closed-loop response from sensor data to the canard-servo output. Bang-Bang, various aggression simple proportional control, and linear-ramping setpoint proportional controllers were tested. The final design implemented a medium-aggression kP and kI PI Controller using a linear-ramping setpoint from the measured roll to a final target roll of 0.

EXPECTED OUTCOMES AND TECHNICAL IMPACT

Ground testing has validated mechanical alignment tolerances, servo command fidelity, and structural robustness under simulated aerodynamic loads. The system demonstrates rapid actuation response and alignment accuracy within the required tolerances for roll authority. During flight testing at the 2025 IREC competition, the system is expected to demonstrate stable, low-latency roll control throughout ascent, with self-stabilizing behavior in fault scenarios. The development of this tri-canard system reflects a synthesis of mechanical ingenuity, aerodynamic sensitivity, and regulatory compliance, offering a template for scalable roll control in high-performance suborbital vehicles. As such, it represents a meaningful contribution to the field of student-led rocketry and advanced control system design.

Design and Development of a High-Performance Rocket Nozzle with Parabolic Geometry Defined by a Multi-Parameter Rao Method Application and Optimized Structural Integration

Team 401 Project Technical Presentation to the 2025 IREC

Gabriel S. Plata and José Roberto S. Carmo

Universidade Federal de Santa Catarina, Joinville, Santa Catarina, 89219-600, Brazil

I. Introduction

Developing a rocket nozzle with an ideal internal geometry coupled with a inexpensive, sound and lightweight structure is one of the greatest challenges encountered in the design of solid propellant rocket motor. Since the conception of the MORPHEUS project, it was realized that, given the restriction on the use of a low-efficiency propellant, developing a design to reach the 30,000 ft mark would necessitate a lightweight and highly efficient motor. The efforts dedicated to achieving a geometry that maximizes thrust without adding significant complications to the project's execution, while simultaneously meeting the parameters of our propellant, compelled us to develop and implement a multi-input method based on Rao's geometry theory. In conjunction with the internal geometric development, relatively low-cost yet high-performance materials of ablative and/or very high melting point character were implemented so that a significant reduction in structural mass could be tangible.

II. Design and Efficiency

Among the initial steps in developing an efficient rocket motor is the ballistic analysis, which enables the first CAD iterations of the project. This process typically loops back to the ballistic model, repeating until a balance between structural integrity and performance is achieved. The choice of a bell nozzle is directly linked to the efficiency requirements of the system. However, the input data used to define its profile depends heavily on the ballistic model, which undergoes multiple iterations before stabilization. If not properly handled, this can turn the nozzle efficiency into a variable parameter, undermining design consistency. This highlighted the need to parametrize the parabolic geometry of the nozzle based on ballistic data, allowing its efficiency to remain constant throughout design iterations while ensuring a practical, fast, and accessible implementation.

The Rao method was developed empirically to generate the parabolic contour of bell nozzles, based on input parameters that are often difficult to obtain during the early stages of rocket motor design. Consequently, adapting Rao's methodology to the preliminary design of an experimental sounding rocket motor became a key objective for the team. By leveraging thermodynamic analysis and flow condition assumptions, alternative input relationships were established to serve as equivalents to those originally required by Rao, while still yielding identical final parameters. For instance, operational chamber pressure and desired exhaust pressure (readily available from ballistic simulations and propellant evaluation programs) were used in place of Mach number data, which typically requires a completed nozzle geometry to be determined through CFD or testing. These equivalences enabled the use of reliable, easily obtainable inputs to generate a nozzle profile that would remain largely stable throughout the project's development, maintaining consistent efficiency across design iterations.

Then it became possible to determine all parameters required for implementing the Rao method using only four basic inputs, readily available during the early design stages: nozzle throat radius, specific heat ratio of the propellant, average chamber pressure, and desired exit pressure. The method begins with the construction of an equivalent conical nozzle designed to meet the same pressure ratio requirements. This conical profile serves as a geometric reference for the subsequent generation of the bell contour. The conical nozzle is not used as the final design, but rather as a scaffolding from which the parabolic bell profile is derived, ensuring a smooth and optimized expansion. As a result, two additional

parameters are introduced: the divergence half-angle of the conical nozzle and the percentage of the bell nozzle's total length relative to that of the equivalent cone. These values allow for accurate geometric interpolation, guiding the transition from the idealized cone to the high-efficiency bell contour prescribed by Rao.

Once all analytical parameters had been defined, the next challenge became generating the nozzle's parabolic contour. This profile can be modeled using a quadratic Bézier curve, where the primary difficulty lies in determining the tangents at the start and end points (prior to curve generation) and using their intersection to define the control point. To address this, Rao conducted multiple hot-fire tests with nozzles of varying divergence angles and relative bell lengths, eventually correlating the contour angles to the nozzle's expansion ratio and normalized length. Reconstructing this empirical data with sufficient accuracy from historical diagrams motivated the development of a Python algorithm capable of interpolating between known data points from these original functions. This interpolation allows for precise determination of the Bézier control point, complementing the already analytically defined start and end points.

With these three points established, a two-dimensional nozzle profile is generated, which, when revolved about a chosen axis, yields the internal geometry of the bell nozzle. The implemented method was validated through CFD analyses, which demonstrated that the resulting geometry provides superior efficiency compared to conical nozzles and other bell nozzle profiles generated using alternative design methods.

III. Mechanical Integration and Materials

To meet the initially defined design constraints: low cost, high thermal resistance, low mass, and a high degree of reusability, specific materials were carefully selected for each section of the nozzle. The outer shell was constructed from H13 tool steel, a material commonly used in hot working applications. H13 offers a favorable balance between cost and performance, presenting superior yield strength compared to high-strength stainless steels, while maintaining good machinability and high-temperature resistance. For the thermal protection layer, Celeron was chosen, a phenolic composite made from cotton fabric and phenolic resin. This material combines a low density of 1.35 g/cm^3 with high specific mechanical strength and an exceptionally low thermal conductivity. Critically, Celeron exhibits ablative behavior: as hot combustion gases flow over its surface, the material undergoes controlled carbonization. The resulting fragile carbon layer is gradually eroded, carrying away surface heat through material removal, thus preserving structural integrity beneath.

For the nozzle's throat, the most thermally and mechanically stressed region of the entire propulsion system, subjected to extreme gas velocity, temperature, and pressure gradients, a solid block of graphite was selected. This block was machined to the previously defined throat geometry and was chosen primarily for two reasons: its extremely high melting point and its ability to retain mechanical strength with minimal degradation under elevated temperatures. While inherently brittle under mechanical shock, the material offers sufficient compressive and tensile strength to endure the intense thermal and fluid dynamic loads of the throat region without structural compromise.

The mechanical integration strategy prioritized a reliable and rapid assembly method. Notably, the component responsible for attachment to the motor casing was designed as a separate element from the main nozzle structure. High-temperature red silicone will be implemented both as a sealant, ensuring a secure and thermally resistant bond between the H13 steel outer shell, the graphite throat insert, and the Celeron thermal protection layer, and contributing to the structural integrity of the nozzle assembly. This decoupled attachment design offers significant advantages, including enhanced adaptability for integration with potential future nozzle designs utilizing the same combustion chamber. Furthermore, this approach facilitates weight reduction, as the separate attachment component can be manufactured from a lighter material than H13 steel, such as aluminum, without compromising the structural integrity of the nozzle itself.

IV. Conclusion

The methodology developed for the MORPHEUS nozzle proved effective in generating a high-efficiency bell profile from early-stage design inputs. By adapting Rao's method and integrating it with a Bézier-based algorithm, a stable and optimized geometry was achieved. CFD analyses demonstrated superior performance over conical and conventionally designed bell nozzles. The use of ablative and high-temperature-resistant materials, combined with a modular and lightweight structural design, ensures mechanical integrity and thermal protection. These results support the reliability of the approach and its potential for successful flight performance at the Intercollegiate Rocket Engineering Competition.

V. References

- [1] R. Newlands, “The Thrust Optimised Parabolic Nozzle,” Apr. 2017.
- [2] NASA, Liquid Rocket Engine Nozzles, NASA SP-8120, Washington, D.C., July 1976.
- [3] G. V. R. Rao, “Exhaust Nozzle Contour for Optimum Thrust,” *Jet Propulsion*, Vol. 28, No. 6, 1958, pp. 377–382. <https://doi.org/10.2514/8.7156>
- [4] R. Nakka, “Theory of Nozzles,” http://www.nakka-rocketry.net/th_nozz.html [Accessed Dec. 2023].
- [5] K. R. Sreenath and A. K. Mubarak, “Design and analysis of contour bell nozzle and comparison with dual bell nozzle,” P. G. Scholar and Associate Professor, Govt. Engineering College Thrissur, Kerala.
- [6] MatWeb, “AISI H13 Hot Work Tool Steel,” <https://www.matweb.com/search/datasheet.aspx> [Accessed 2025].
- [7] Y.-l. Wang, K.-x. Song, Y.-m. Zhang, and G.-x. Wang, “Microstructure Evolution and Fracture Mechanism of H13 Steel During High Temperature Tensile Deformation,” *Materials Science and Engineering: A*, 2019.
- [8] C. Wang, W. Tian, and K. Zhang, “Thermal Structure Strength Analysis of Nozzle of Solid Rocket Motor with the Coupled Algorithm,” *International Journal of Aerospace Engineering*, Vol. 2021, Article ID 6653824, 11 pp. <https://doi.org/10.1155/2021/6653824>
- [9] W. B. Hall, Standardization of the Carbon-Phenolic Materials and Processes, Volume I: Experimental Studies, Final Report, NASA Grant NAG8-545, Mississippi State University, prepared for NASA George C. Marshall Space Flight Center, Aug. 31, 1988.

Modeling and Design of a Custom Radio for Telemetry and Live Video

Team 405 Technical Presentation to the 2025 IREC

Joseph B. Hauerstein
University of Maryland, College Park, MD 20742, USA

INTRODUCTION

Radio transmission of telemetry is often used in high-power rocketry to assist in both tracking and recovery operations. Due to advances in video encoding algorithms, such systems also present the opportunity for the transmission of digital video. The Terrapin Rocket Team at the University of Maryland, College Park incorporates multiple radio transceivers into their 2025 competition rocket, Helios, for the IREC. These transceivers form a link with a ground station to provide telemetry and video during the flight. Limited full-scale test flights and the challenging RF environment at the IREC present the need for modeling radio performance to ensure link closure. A link analysis was performed in MATLAB to set hardware and software requirements. A radio module was designed to meet these requirements using commercially available ICs, and circuit topology was verified using LTspice to minimize electrical losses. Reed-Solomon forward error correction (FEC) was implemented to ensure accurate reception of transmitted data.

LINK ANALYSIS

A link analysis was performed in MATLAB using the RF and Communication toolboxes. The analysis consists of three major components. The first is the calculation of received power, which models the signal travelling through space from the transmitter to the receiver. The second is the calculation of sensitivity link margin, representing the ability of the radio to detect the presence of a signal. The third is estimation of bit error rate (BER), or the ability to accurately distinguish the signal from noise.

Received power was calculated first by computing the free space path loss and the polarization loss (due to antenna misalignment). Path loss was computed for the 70 cm band (420 MHz – 450 MHz), which was selected because of low path loss and familiarity with operating telemetry radios in the band. Distance and angle between antennas was calculated over the course of a flight using data collected from the Terrapin Rocket Team's 2024 flight at the SAC. The altitude data was multiplied by 3 to reflect the change in target altitude from 10,000 ft in 2024 to 30,000 ft in 2025. This produced a data set accurate enough to estimate the largest loss that would be experienced during the flight. Hardware characteristics were then applied to predict the power at the input to the receiver. These include antenna efficiency, antenna gain, antenna polarization, and transmit power. Antenna efficiency and gain were estimated based on current hardware, and three combinations of antenna polarization were considered: linear-linear, linear-circular, and circular-circular. The calculated received power was subtracted from the sensitivity of a representative radio IC to determine sensitivity link margin at various bitrates.

The received power was then used to compute signal to noise ratio (SNR) to estimate BER for frequency shift keying (FSK) modulation. This modulation was selected due to wide availability of radio ICs and analytical expressions being available for FSK BER as a function of energy per bit to noise spectral density ratio (E_b/N_0). SNR was computed by summing the expected noise figure for the PA and RF switch with the noise temperature of the system at the standard of 290 K, including some margin. E_b/N_0 was computed for bitrates of 9.8 kbps, 500 kbps, and 1 Mbps, which were used to set signal bandwidth. BER was calculated for 2-FSK and 4-FSK over additive Gaussian white noise (AWGN) and Rician fading channels. The AWGN case represents the best case scenario, while the Rician fading case represents the worst realistic case, with multipath signals having the same power as the line of sight signal. Assuming pure random error, the expected number of errors per packet was computed using this data. Reed-Solomon FEC was applied to randomly generated data in MATLAB, and the estimated BER was then applied to determine the required code rate to ensure effectively zero received errors.

HARDWARE DESIGN

The Silicon Labs Si4463 radio IC was selected to meet the hardware requirements from the link analysis, namely 70 cm band operation, FSK modulation, 500 kbps bitrate, and better than -90 dBm sensitivity at this bitrate. It also contains an automatic frequency control (AFC) block that can be used to compensate for doppler shift. In order to meet the +33 dBm transmit power requirement, an external PA was required. A switched RX/TX topology was selected for the RF section to allow for the same antenna to be used for both receive and transmit. A linearly polarized dipole antenna was selected for the rocket due to space constraints. This dipole will interact with other conductive components in the rocket, namely the metallic and carbon fiber components, causing interference with the antenna. This effect must be corrected to achieve acceptable efficiency and avoid possible damage to the radio. To correct for this effect, a custom dipole was tuned to resonate correctly despite interactions with other conductive components. A right hand circularly polarized (RHCP) Yagi-Uda antenna was selected for the ground station. This linear-circular polarization combination was found to be the most advantageous because it results in a consistent 3 dB loss. The circular-circular or linear-linear configurations could allow for zero loss, but also allow for much higher losses at specific relative orientations.

A Pi low pass filter was added between all major components in the RX and TX chains to attenuate unwanted harmonics and out of band noise. Filter and matching circuit component values were verified via simulation of the frequency response in LTspice, and tuned for better than -6 dB S11 in the 70cm band. All RF circuits were matched to 50 Ohm impedance. The recommended RX matching circuit was used for the radio IC, and the TX PA and matching circuit for the radio IC were calculated from manufacturer recommendations. External PA impedance at input and output ports was calculated from S-parameters provided by the manufacturer. All digital logic and RF circuits were kept isolated to reduce interference.

SOFTWARE DESIGN

Communication with the radio IC was carried out via SPI on the hardware level, and via the Silicon Labs EzRadioPro API on the software level. A custom implementation of this API for the Si4463 was written in Arduino C++ to allow maximum flexibility in communication with the radio IC. The WDS software from Silicon Labs was used to generate the required configuration files for all required modulation and bitrate combinations.

The internal packet handler was configured for packets up to 8191 bytes in length, with the actual length specified an additional leading two bytes. Long packet length reduces inefficiencies from the preamble and sync words transmitted at the beginning of each packet that would otherwise reduce usable bitrate. To support packet lengths greater than the 129 byte maximum FIFO buffer length, the packet transmission logic was constructed to allow for dynamic refilling of the radio IC's internal FIFO buffer during packet transmission. This configuration provides for a number of additional efficiencies, including the ability to begin sending a block before the FEC codes are generated, and then add the codes to the end of the block once the microcontroller has all the data in the block. As stated in the link analysis, Reed-Solomon FEC was used to ensure accurate reception of data. Based on the link analysis, a minimum 10 check symbols are required to receive packets in the worst case, resulting in 245 bytes of data for each 255 byte block, or a (245, 255) code. At the ground station, packets are transferred to a laptop which processes the video and telemetry data.

RESULTS, CONCLUSIONS, AND FOLLOW-ON WORK

A radio was designed for the transmission of telemetry and video to meet link requirements for the flight of Helios at the 2025 IREC. These requirements include a 70 cm band operation, FSK modulation, 500 kbps bitrate, better than -90 dBm sensitivity, +33 dBm transmit power, omnidirectional linear rocket antenna, >6 dBi RHCP ground station antenna, and (245, 255) Reed-Solomon codes. These requirements were found using a link analysis performed in MATLAB. RF circuit topology and component values were verified by simulating the frequency response in LTspice. Radio prototypes were flown on sub-scale and full scale test flights, though most flights failed to collect meaningful data due to hardware issues with the prototypes. The full radio design was ground tested over 1/3 of a mile, resulting in a signal strength of around -80dBm. These results were as expected given the hardware and environmental conditions. Results from the test flight at the IREC will be compared with simulated and test data to verify hardware requirements and accuracy of the model used.

Design, Development, and Testing of a Telemetry Agnostic Ground Station

Team 407 Project Technical Report for the 2025 IREC

Kuba J. Kedzior*

University of Minnesota Rocket Team, Minneapolis, MN, 55414

University of Minnesota Rocket Team

University of Minnesota, Minneapolis, MN, 55414

INTRODUCTION

Team 407, University of Minnesota, has developed a cross-compatible ground station software application, *Wings*, designed to interface with various commercial and custom telemetry systems. The software supports multiple telemetry inputs without requiring modifications to its codebase, improving the team's awareness of its rockets' state during launches. Historically, the team has preferred to use various *Entacore Electronics*¹, *Altus Metrum*², and *Featherweight Telemetry*³ systems, as well as a custom SRAD flight computer. To support these missions, the team requires multiple ground stations for complete telemetry coverage. *Wings* consolidates these sources into a unified interface, significantly improving awareness of the ground crew during flight operations and recovery. The system has been validated through unit testing, ground tests, and live flight applications.

DESIGN

Wings was developed with the *Tauri app development framework*,⁴ allowing for a web-based frontend as well as a backend written in Rust. This architecture enables the team to use Macintosh and Linux systems as base stations in place of commercial applications that would only be available on Windows.

To receive telemetry from AIM-XTRA flight computers, the team collaborated with Entacore Electronics to create a Rust-based HID adapter based on documentation and guidance provided by the company. Serial communication was also established for Altus Metrum and Featherweight Telemetry Receivers using public documentation. In addition to commercial telemetry systems, the ground station communicates with the team's flight computer via an RFD900ux2 radio unit and can accept pre-generated data from simulations in CSV format. Using the strategy design pattern, *Wings* can dynamically accommodate each telemetry protocol at runtime, enabling simultaneous communication with multiple devices. Due to the low throughput of hobby rocketry telemetry systems, it was found to be sufficient to handle communication on a single thread. However, the strategy design pattern has made the codebase accessible for future multithreading I/O operations.

In addition, the user interface has been developed to fully support different mission requirements and telemetry formats. Before flight, operators configure the mission's expected telemetry sources and desired visual components via a JSON configuration file populated through the settings menu UI. *Wings* supports multiple display types, including multiple types of plots, orientation indicators, textual readouts, and status lights. The ground station has also been written with a documented software interface to ease future development and expansion. Operators can define the location, size, and data for each element on the screen to best suit their mission.

Unlike the commercial telemetry software solution, which requires users to switch between each vendor's software, *Wings* centralizes all telemetry streams into one application. This enables real-time cross-referencing of critical metrics, such as descent velocity and landing location. In addition, live telemetry can be displayed alongside predicted data to assess flight performance and recovery system function, allowing for improved comprehension of vehicle state throughout the mission.

* Software Lead, UMN Rocket Team, kedzi011@umn.edu

TESTING & PREVIOUS USE

To ensure robust data handling, Wings underwent extensive unit testing of its parsing algorithms and backend communication layers. Unit tests were written in Rust using the built-in testing framework, with coverage targeting parsing edge cases, protocol mismatches, and malformed packets. These tests were run repeatedly and expanded upon throughout the development process to ensure that all edge cases were covered. Additionally, the application was also written to log all raw data into binary files before attempting to parse it, preserving data fidelity and allowing for reprocessing in the event of a parsing failure.

To ensure reliable data acquisition from the receiver hardware, the ground station was used in 30+ successful ground tests of varying length, gathering telemetry for up to 24 hours. These tests confirmed compatibility with all supported receivers and verified consistent data acquisition.

The application has also been used as a remote data acquisition system during its 2025 IREC motor static fire, as well as a ground station for its 2024 high-altitude launch and 2024 IREC recovery system validation flight. During the static fire, acquired data was broadcast to a secondary laptop running Wings. Wings was configured to instantaneously identify the impulse and max thrust of the motor, allowing it to generate a motor code⁵ from the acquired data during the static fire. In flight applications, the software successfully gathered data from both commercial and custom avionics systems.

CONCLUSION

Wings provides the team with unified, real-time telemetry monitoring through a single customizable interface. The system has demonstrated reliable performance in both ground and flight conditions. During the 2025 IREC competition, the system aims to aggregate data from 4 redundant sources, providing a large-scale test of its potential as a consolidated point of reference during flight.

REFERENCES

- ¹Carthy, A., “Entacore,” *Home | Entacore Group*, Pretoria, South Africa, Available: <https://entacore.com/>.
- ²“Altus Metrum,” *Altus Metrum* Available: <https://altusmetrum.org/>.
- ³“Featherweight altimeters,” Featherweight Altimeters Available: <https://www.featherweightaltimeters.com/>.
- ⁴“Tauri 2.0,” *Tauri* Available: <https://tauri.app/>.
- ⁵National Fire Protection Association, *NFPA 1125: Code for the Manufacture of Model Rocket and High Power Rocket Motors*, NFPA, Quincy, MA, 2022.

SRAD Carbon Fiber Composite Airframes

Team 409 Project Technical Presentation to the 2025 IREC

Jonah Lin, Jacob Sawyer, Trent Brown, and Gavin Lewis

University of Washington, Society for Advanced Rocket Propulsion, Seattle, WA, 98926, United States

Introduction

This year, the Society for Advanced Rocket Propulsion (SARP) aimed to improve the speed, quality, and environmental safety of our student-researched and developed (SRAD) carbon fiber airframes. Previously, SARP's airframe manufacturing relied on expensive, precisely machined mandrels that constrained the diameters of future rocket projects. Additionally, previous manufacturing required the mandrel to be coated in chemical sealants and release agents that added hours to the layup time and were severely toxic. Furthermore, old SARP airframes were significantly overbuilt for their use cases due to uncertainties in the composite material mechanics. SARP tackled these challenges in a two-fold approach—the first is a new airframe layup process, making greater use of plastic release films and student-built tooling to decrease the average production time by roughly 25%. Second, the team implemented ASTM standard composites testing to optimize ply counts and improve airframe structural efficiency.

SRAD Methodology and Changes

SARP has long emphasized the importance of nearly full student control for composite structure design and manufacturing. SRAD composite components not only improve the design and manufacturing skills SARP members gain but also provide significant design freedom for other rocket subsystems. We have constructed our own carbon fiber, custom composite fins, and nose cones for the last three consecutive years. SARP's experience and generous donations of materials from local manufacturers make producing custom composites significantly more cost-efficient than outsourcing or purchasing from a commercial supplier. Finally—and most importantly—customization of our composites gives us flexibility in design, allowing us to alter the size, number of plies, and ply orientation to fit the requirements of our rocket.

Manufacturing

Manufacturing of the airframes starts with the setup of the alignment jig. The alignment jig is a large plywood board with two parallel lengths of angle iron fixed on opposite edges, 5 feet apart. A roughly 5-foot by 5-foot plexiglass sheet is placed squarely between the angle iron and on top of a layer of breather. Tape marks the edges square to the jig's rails for alignment later. A layer of release film is taped to the top surface of the plexiglass.

Next, the release film is tightly wrapped around a 59-inch-long, 6-inch-diameter 6061 aluminum tube stock, used as a mandrel. This process is quite quick and much less toxic, removing the need for a slow-setting, hazardous chemical release. These layers of release film allow for a variable inner diameter tolerance that can be changed to fit various components. 3D-printed end caps are placed at the ends of the mandrel and integrate into angle iron sections on the alignment jig. Altogether, the jig and end caps create an on-axis roll that would be challenging by hand.

A CNC fabric cutter cuts out 5 plies of a woven bidirectional prepreg carbon fiber. The length of the plies are cut with an additional inch, compensating for any errors during manufacturing. The airframes have a ply order of $[0^\circ/\pm 45^\circ/0^\circ/\pm 45^\circ/0^\circ]$, resulting in an airframe with quasi-isotropic properties. Before layup onto the mandrel, the backing on both sides of the first ply is removed, though one side is replaced. With this side down, the carbon has better adhesion to the release film-wrapped mandrel instead of the alignment jig (also release film-lined). Pre-peeling the backing keeps the ply from peeling off the mandrel when preparing for the next ply. For each subsequent ply, the backing on both sides can be peeled since the adhesion to the previously rolled ply is greater than the release film's. After peeling the first side, each ply is placed facedown on the jig, and lined up on the alignment markings. The plastic backing is then removed. Notches on the mandrel's 3D printed end caps and jig are

designated starting orientations, spacing the seams radially equidistant from each other. Next, the mandrel is rolled over the ply, applying a constant downward force to ensure adhesion. This process is repeated for each ply until all 5 plies have been placed. The exposed release film is cut off, leaving room for the sealant tape to adhere to the mandrel during bagging. A layer of release film and breather is then added to cover the layup, and an additional flap of breather is added for the vacuum port to sit on.

Next, bagging film is cut to size and sealed around the mandrel with bagging tape, enclosing the airframe. A 5-inch pleat spans the length of the airframe and holds the vacuum port. The bag is placed under vacuum and checked for leaks. If less than 5 inHg of vacuum is lost in 10 minutes, the bag is deemed “sealed” and will hold vacuum during the cure. At optimal efficiency, the total layup time before cure is roughly 1.5 hours. The airframe is then placed into the autoclave on breather-wrapped carbon steel stands to protect the vacuum bag. The autoclave runs the 6-hour cure cycle. This process is run for all airframes, though multiple airframes can be cut out of a single longer cure, reducing the total manufacturing time even further.

Once the airframe is removed from the mandrel, post-processing can start. Using a cutting wheel on a Dremel, the airframe is cut to a ¼ inch longer than the desired length to remove any imperfections from the cure or layup. A COTS machined part, such as a close-fitting fiberglass coupler, is then clamped into or around the airframe and used to guide the final sanding of the airframe, shortening it to the correct size and finish. This is repeated on both ends of the airframe. Using low-grit sandpaper and razor blades, the bonded surfaces are scratched or gouged, fully preparing the surface for epoxy or filming adhesive. Finally, a 3D-printed jig is used to mark and drill holes for integration with other components. In total, all post-processing can be done in about 2 hours per airframe.

Testing

The carbon fiber of the airframes was analyzed with a variety of coupon tests. The major purpose of the testing was to help us determine between $[0^\circ/\pm 45^\circ/0^\circ/\pm 45^\circ/0^\circ]$ ply orientation or $[0^\circ/\pm 45^\circ/0^\circ/0^\circ/\pm 45^\circ/0^\circ]$ ply orientation. Following ASTM D3039, a tensile test on coupons of both orientations was completed, to test the most probable failure points of the airframes, the bolt holes, and epoxy connections. To test the bolt holes, a coupon with a ¼ inch hole was placed into the lower grips of the Instron with a ¼ bolt connecting it to a jig secured in the upper grips of the Instron machine. The Instron machine then pulled until failure, recording the force applied. The results showed that 5 plies is strong enough to survive the tensile loading with a sizable factor of safety. For the epoxy testing, the carbon fiber coupon was epoxied to an aluminum coupon, varying the area of overlap between test subjects. Spacers were added to ensure that only axial stress was applied to the coupon. The force was recorded and divided by the epoxied surface area to determine the shear stress. It was found that the current method of preparing the carbon surface and the type of epoxy used would be strong enough for the expected forces.

Results, Conclusions, and Follow-on Work

For the final components, the airframes ended up being slightly non-concentric. SARP determined this was due to the less precise mandrel. In the future, the mandrel could be machined for a better surface finish, concentricity, and circularity, though it isn't feasible on university machinery and would have to be outsourced. The full effect of this non-concentricity is still being evaluated, but all integration was successful despite it. Otherwise, the airframes were completed to an acceptable standard, comparable to previous years.

The main improvements were made in the manufacturing. All of the airframe manufacturing was completed in 6 days spread out over the course of two weeks—a team record. Improvements in process streamlining, parallel operation, and release choices reduced the layup time itself from a 5+ hour process to an under 2-hour process. Significant time can still be saved in the curing step too. While the cure itself cannot be easily changed, the requirement reduction of the mandrel can allow for multiple mandrels and multiple cures to be executed in the same autoclave cycle. Tooling is currently being developed to facilitate this process, and overall, it would reduce the cure time per airframe even further, effectively cutting it in half. Overall, the SRAD airframes have been a major success in achieving our team's goals and will continue to be a core component of our SARP's program.

Design and Process Development of a Multi-Shot CFRP Fincan

Team 601 Project Technical Presentation to the 2025 IREC

Leonard Weiß, Louis Faber, and Julian Waller
WARR, Garching, Bavaria, 85748, Germany

Introduction

The EX-1Evo represents WARR Rocketry's latest two-stage sounding rocket and is the consistent further development of the EX-1E, WARR's 2024 SAC vehicle. The primary focus of the development is on structural and aerodynamic optimization, especially in the fin sections. Whereby the fins are often bolted, bonded, or riveted to the fuselage in simpler builds, the whole fincan can also be manufactured as one integral part, using a tip-to-tip layup. Even though the latter already delivers quite good results, we decided to use an even more sophisticated approach with a multi-shot CFRP fincan, consisting of four quarter sections, two ribs, and one reinforcement ring. Further, the sustainer fincan was equipped with a precision-machined staging interface. The structural and aerodynamic design was performed with various numerical tools to optimize flow and stress conditions in flight regimes up to Mach 2.5.

Aerodynamic Design

The EX-1Evo's sustainer fincan is responsible for the aerodynamic stabilization of the upper stage in flight, since the whole vehicle is passively stabilized. Stability in this context refers to static stability, which describes the reaction of the rocket to a disturbance. If the disturbed state imparts a restoring moment to the rocket, meaning it returns to the initial state, the rocket is statically stable. This is achieved if the aerodynamic center of pressure is located behind the center of gravity. With a wingspan of 19.53 inches, the stability margin of the sustainer reaches a minimum with 2.50 calibers, fulfilling the stability requirement of more than two calibers. The stability after stage separation is 2.75 calibers.

The reduction of drag is the second important design driver in order to reach the target apogee of 45.000 ft. The leading edge is swept to 65°, which results in a subsonic leading edge up to Mach 2.36; afterwards, the sharp leading edge helps to reduce drag.

Structural Concept

The outer layers, and thereby the contour of the fincan, consist, just as in a tip-to-tip layup, of CFRP sheets. The difference, however, lies in their manufacturing method. Instead of wet laminating over the existing fin geometry (be it an actual, load-bearing fin or a foam core), the outer sheets were produced separately.

Therefore, the fincan geometry was split into four identical quarter shells, and a mold from PEI-CF was 3D-printed, CNC-milled, and sanded to the final dimensions. This allowed the use of CYCOM 977-2A structural CFRP-Prepregs, cured according to the datasheet cycle in an autoclave, for laminating the shells. Two stripes with GFRP sacrificial plies were added on the inside to support the ribs' structural integrity. The resulting four individual quarter sections were then milled to shape to obtain a precise contour.

The stiffness was further increased, using two ribs, cut out of 0.2-inch-thick CFRP plates. Those ribs are star-shaped; therefore, the individual fins are structurally interconnected, significantly increasing the stiffness. Those ribs feature a circular cutout in the middle, which guides and centers the motor casing in the fincan. Together with the reinforcement ring and the ribs, the four quarters were bonded with LOCTITE EA 9394 AERO structural epoxy. Forged carbon edges were glued to the sustainer's leading and trailing edges, aiming for superior aerodynamics and rigidity. The alignment of the ribs inside the quarters is ensured by CNC-milled grooves in the GFRP sacrificial plies. To align the four quarters with the leading and trailing edges, a 3D-printed assembly jig was used.

After the CFRP structure of the fincan was finished, aluminum and titanium rings were bonded to it. Those were then milled to their final dimension and tolerances in a subsequent step, ensuring the coaxiality of both fitting surfaces for the staging mechanism. The lower aluminum ring also serves as a thrust insertion, providing the best possible motor alignment.

Dimensioning and Simulation

The following loadcases are expected and were therefore simulated:

- Aerodynamic lift of 191 lbf per fin

- Thrust loads of 1214 lbf
- Integration/erection bending moments of 406 ft*lb

The FEA (Finite-Element-Analyse) of both fincans was done using Ansys, with its ACP (Ansys Composite Prepost). The calculated aerodynamic loads onto the fins and the structural loads on the metal inserts were applied, respectively. A dedicated mesh was assigned to every component; most interestingly, the shells had a mesh size of 0.2 in. Afterwards, a layup for the shells was created where the upper stage has four layers of CFRP, whereby only one layer spreads across the whole wingspan. The thickness of the CFRP is assumed to be 0.014 in, as former tests have shown. The fincan features two layers of $0^{\circ}/90^{\circ}$, whereby only one layer is the outer surface of the fins and the fuselage, the second only supports the latter.

Furthermore, the upper stage fincan features two $\pm 45^{\circ}$ layers going only roughly two inches into the fin. The final layup in the fuselage section of the fincan is $0^{\circ}/90^{\circ}|\pm 45^{\circ}|\pm 45^{\circ}|0^{\circ}/90^{\circ}$. The most critical loadcases were the aerodynamic forces in flight. The resulting deformation is below 0.1 in. The minimum safety factor for the fincan is 4.7.

Autonomous Parachute Guidance System

Team 700 Technical Presentation to the 2025 IREC

Gabriella Best, Trevor Dales, Thomas Rimer

Cornell University, Ithaca, New York, 14853

Introduction

The Cornell Rocketry Team's "Brake Line Manipulation System" (BLiMS) is a parachute guidance system designed to autonomously steer our 19 foot, 150-lb launch vehicle to a predetermined ground waypoint during the descent phase of a rocketry launch. BLiMS uses a pair of counter-rotating spools to extend and retract the brake lines of a ram air parachute, altering the aerodynamic profile, allowing for control over heading and glide slope. An onboard 20Hz GPS feeds location and altitude data to a Raspberry Pi Pico microcontroller running a custom closed-loop feedback control algorithm, which constantly adjusts the brake lines to match the desired heading. Through multiple test flights, we have refined BLiMS' ability to execute precise steering maneuvers, demonstrating its ability to autonomously guide a high-powered rocket using real-time GPS data.

Parachute

BLiMS uses a 110 square foot rectangular parafoil that provides both lift and steerability—fundamentally different from the uncontrollable round parachutes conventionally used in amateur rocketry. Extending or retracting brake lines connected to the edges of the parachute adjusts the glide ratio and bank angle of the parachute, allowing for pitch and yaw control similar to that of a skydiver. However, the increased functionality of rectangular parafoils also makes them more sensitive to packing; line twists and improper stowing easily leads to asymmetrical inflation or even total deployment failure. We therefore worked closely with skydivers throughout development to refine the parachute packing, including drop tests from skydiving planes to document inflation dynamics up close.

Mechanism

At the heart of BLiMS is a single, BLDC motor which spins a pair of counter-rotating spools on which the brake lines are wound. During normal operation, each brake line can be adjusted 1 foot above or below half-brake (nominal), allowing for both small turns and extreme cork-screw banks. A worm gear coupling the motor to the spools prevents high forces experienced during initial parachute deployment from backdriving the motor. A pair of student-CNC'd aluminum bulkheads provide mounting points for the parachute's static lines, with loading capabilities up to 4,500 pounds—verified by a specially built loading rig. The whole system was simulated extensively with commercial FEA tools as well as a custom, student-written Python simulation to identify appropriate motor speeds and gear ratios.

Software

BLiMS uses a barometric altimeter and GPS to autonomously navigate to a predetermined GPS coordinate sent to the rocket prior to flight via our student-developed ground station. After a set delay following main parachute deployment, the rocket's onboard flight software running on a Raspberry Pi Pico executes a method that initiates BLiMS's flight sequence. In addition to controlling the motor, the flight software also relays all commands back to a ground receiver via a radio allowing for remote monitoring of the BLiMS controller.

Feedback Control

The BLiMS controller operates by continuously aligning the system's track vector—the direction of actual travel—with the bearing vector pointing toward a predefined target. At each timestep, GPS data is used to calculate both the current track and bearing angles. The difference between these two angles defines the steering error, which is fed into a PI controller that outputs a motor angle command. With this control scheme, the accumulating integral term effectively trims the parafoil's flight, enabling the controller to compensate for unpredictable factors such as

wind. A downwind waypoint and conservative tuning of the controller further contribute to the robustness of BLiMS against crosswinds.

Preliminary Results

To characterize and tune BLiMS, we developed a dedicated L3 rocket—independent of our final launch vehicle—as a test platform. Throughout the winter, we successfully flew BLiMS three times on the L3 test bed in addition to one flight on our actual launch vehicle. These tests provided a controlled environment to incrementally validate the performance of the system under actual flight conditions without risking damage to our competition launch vehicle. Through these trials, we refined both the hardware and software integration of the BLiMS mechanism and were able to gather valuable data to inform our control logic.

Across these launches, BLiMS was successfully activated in two flights, each demonstrating its ability to steer the parafoil during descent. In one test, we executed a predefined turning sequence that clearly produced directional changes in the rocket's descent path, validating our mechanism functionality. In another, we implemented an improved turning logic that captured a broader range of dynamics, enabling us to gather comprehensive data for tuning the controller. GPS data was successfully recorded during these flights, and our post-flight inspection confirmed that the mechanical components operated as expected. Additionally, we made design improvements between launches—such as adding weight to the nose cone—to better simulate final flight conditions and ensure stable flight dynamics. These test flights allowed us to comprehensively characterize the dynamics of our descent trajectory; we also learned a lot about how wind can unpredictably affect the descending rocket.

The data collected during these launches demonstrates BLiMS' ability to alter descent behavior in a controlled and predictable way. Visually, the descending rocket clearly executed the alternating turns we commanded. The logged rotational velocity of the rocket shows a clear and consistent response to BLiMS motor inputs, confirming that the system effectively induces controlled turns. This relationship not only confirms BLiMS's steering capability but also provides crucial insight into the system's dynamics—specifically, how a given motor input translates to a measurable turn rate. With this understanding, we selected concrete gain values for the controller that will produce the smooth, predictable turns required for a stable guided descent.

Conclusion

BLiMS has demonstrated its ability to steer a parafoil using real-time GPS feedback, with successful flight tests confirming that motor inputs produce consistent, measurable changes in the rocket's descent path. These preliminary results have allowed us to characterize the system's turning response and select controller gains that yield smooth, stable maneuvering—critical for accurate targeting while avoiding unpredictable and unsafe behavior. As we prepare for flight at the International Rocket Engineering Competition, we are confident that BLiMS can autonomously guide our rocket to a designated landing zone, meeting both competition requirements and broader goals for precision recovery. The development of this technology enhances the reusability and recovery efficiency of our launch vehicle and holds potential for broader applications such as autonomous payload delivery or aerial survey deployment. Looking ahead, Cornell Rocketry aims to further advance the system by refining and validating the current control scheme, with the goal of making it faster, more accurate, and more resilient to wind disturbances. Continued modeling, along with additional flight testing and data collection, will be essential to achieving these improvements.

Active Braking and Control System for a Sounding Rocket

Team 701 Project Technical Presentation to the 2025 IREC

Anshita Hegde and Rohan S A
Team Abhyuday, IIT Madras, Chennai, 600036

Antony K S and Kartik Agrawal
Team Abhyuday, IIT Madras, Chennai, 600036

INTRODUCTION

Rockets are inherently stable systems that, when designed and manufactured well, can achieve their intended objectives. However, initial designs rely on assumptions – such as ideal flight conditions, perfect manufacturing and optimal performance – that may not hold during the actual flight. Even minor disturbances can impact the vehicle's dynamics and mission performance. To address uncertainties in flight and system imperfections, a robust control system is essential to optimise performance throughout the flight profile.

Building on last year's experimental use of canards for active roll control, developed for the rocket Chetak 2.0 (Spaceport America Cup 2024), the team has made significant progress by developing a comprehensive *Active Braking and Control System (ABCS)*. This innovative system integrates airbrakes and canards to manage both altitude and roll precisely. The primary objective of the ABCS is to ensure the rocket reaches a target of 10,000 feet, compensating for potential discrepancies in simulation data while also maintain a stable roll rate. This is particularly critical for the team's SRAD hybrid rocket engine, which may produce asymmetric forces due to asymmetric thrust or venting, potentially inducing roll and altering the flight path.

Additionally, the team has developed a novel control method and logic for airbrakes and canards, based on simulation and analysis.

ROLL REVERSAL

Simulations of canard deflection at high angles of attack revealed a *roll reversal* phenomenon, where the aerodynamic moment generated by the canard opposed its intended roll direction. This occurs due to sustained vertical disturbances that propagate downstream before fully dissipating, ultimately interfering with the fins. While simulations assumed a perfectly smooth airframe, actual flight conditions – including surface roughness from fasteners, mounts and paint – may promote earlier boundary layer transition, potentially reducing downstream flow interactions.

Further analysis indicates that the canard system disrupts the rocket's expected steady-state aerodynamics, creating a dynamic flow environment. The turbulence generated by the canards travels along the vehicle body, inducing significant flow rotation. When this disturbed flow interacts with the fins, it produces a moment larger than the moment produced by the canards in the opposite direction, enough to reverse the commanded roll direction.

To investigate mitigation strategies, parametric studies were conducted by varying the canards' axial position and their planform geometry. However, these modifications showed limited success in fully decoupling the canard-induced vortices from the fins, suggesting that geometric adjustments alone may be not sufficient.

This highlights the need for robust control algorithms capable of compensating for unsteady aerodynamic effects, as well as higher-fidelity simulations that account for real-world manufacturing tolerances and surface imperfections.

CANARDS CONTROL ARCHITECTURE

The roll reversal phenomenon creates a fundamentally dynamic system that cannot be treated as a steady-state condition. While conventional PID controllers remain an industry standard for many applications, they prove ineffective in this scenario due to their reliance on fixed gain values. The nonlinear unsteady nature of roll reversal, with its sign-changing dynamics and continuously changing reference state, presents challenges that static PID gains cannot accommodate. The controller's inability to adapt to these sign variations leads to persistent and ineffective correction attempts. Furthermore, achieving precise PID tuning would require extensive and costly flight testing to collect sufficient data – a prohibitive requirement for a student-led project of this scale.

To overcome these limitations, the team implemented a *Sliding Mode Controller (SMC)* as an alternative solution. The SMC was specifically selected for its ability to operate effectively in nonlinear regimes and handle sign-changing dynamics without requiring an exact model of the flight characteristics. This model independent approach makes the controller inherently robust against various system disturbances, particularly those encountered during roll reversal events. The SMC's adaptability to unsteady aerodynamic effects and its capacity to maintain stability without extensive tuning make it particularly suitable for this application.

The control system architecture consists of two interconnected loops working in tandem to achieve the desired reference states. The primary controlled parameters are the actuation angle, with a reference state of zero degrees and the roll rate, targeted at 1 radians/second.

The external loop controller receives the error in the desired roll rate as input to generate a commanded control surface deflection. This is then passed to the internal loop, which compares it with the actual control surface position to produce an actuator command. The actuator adjusts the control surfaces which in turn affects the airframe's roll dynamics. The change in sign of the forcing roll moment coefficient derivative during flight indicates roll reversal. The SMC framework adapts to the altered control effectiveness and ensures convergence to a dynamic reference trajectory despite the nonlinearities.

In SMC model, a sliding surface is primarily defined based on the feedback error from each iteration. To ensure stable control, we use a Lyapunov function to verify the system's stability. The controller's parameters are then determined by analysing how the sliding surface behaves. Since SMC naturally abrupt commands that could damage physical components, we smooth these commands using a sigmoidal function. This maintains precise control while protecting the system.

AIRBRAKES CONTROL ARCHITECTURE

The airbrake control system implements a predictive algorithm to precisely manage the rocket's ascent profile and ensure accurate apogee targeting. When the system initially enters the airbrake control phase (5 seconds after burnout), it calculates the rocket's current altitude and velocity to establish the flight state. Upon predicting the apogee and detecting a positive overshoot, the control system activates its predictive estimation routine.

This algorithm performs a polynomial curve fit to model the velocity altitude relationship to an altitude of 10,000 feet and a velocity of 0 metres/second in the vertical axis. The system dynamically selects the optimal polynomial degree for this curve-fitting process to ensure that the fitted trajectory remains below the nominal ascent curve associated with zero airbrake deflection. This ensures that the trajectory remains within controllable bounds.

Using this polynomial model as a reference, the control system continuously compares the predicted velocity values against actual real-time velocity measurements every 10 milli seconds. Any difference between these values generates an error signal quantifying the rocket's deviation from the desired path. This signal feeds into a PI control algorithm, which processes the error to determine the precise airbrake deflection required to correct the flight path.

The resulting control commands actuate the airbrake mechanism, adjusting its deflection angle and modulate aerodynamic drag forces to tune the rocket's ascent energy dissipation.

RESULTS, CONCLUSIONS, AND FOLLOW-ON WORK

The developed Active Braking and Control System (ABCS) presents an innovative approach to addressing critical challenges in rocket flight control. The Sliding Mode Controller (SMC) demonstrates theoretical robustness against non-linear roll reversal effects, while the predictive airbrake system offers precise altitude control through polynomial-based trajectory modulation. These solutions hold significant promise for applications in guided recovery systems and reusable rocket technology. Future development will be focused on enhancing the adaptive control algorithms and validating the predictive models via expanded simulations and small-scale test launches.

REFERENCES

- Mirzaei, M. (2018). Roll reversal phenomenon control in flight vehicles. *Aerospace Science and Technology*, 79, 413–425. <https://doi.org/10.1016/j.ast.2018.05.059>
- Blakelock, J. H. (1965b). *Automatic control of aircraft and missiles*. <https://ci.nii.ac.jp/ncid/BA02286804>
- Wiśniewska, N., Rafalski, A., & Michałek, M. M. (2023). Roll-motion stabilizer design for a sounding rocket using the ADRC methodology. *27th International Conference on Methods and Models in Automation and Robotics (MMAR)*, 490–495. <https://doi.org/10.1109/mmar58394.2023.10242451>
- Zhang, J., Lei, J., Yin, J., & Niu, J. (2020). The numerical investigation on the rolling decoupling of a canard-controlled missile using the jet control system. *Engineering Applications of Computational Fluid Mechanics*, 14(1), 1062–1077. <https://doi.org/10.1080/19942060.2020.1799866>

Ansys® Fluent, Release 2023R1

From the Ground Up: A Full-Stack Architecture to Propel Rocket Avionics

Team 703 Project Technical Presentation to the 2025 IREC

Zeyang Zhang, Maisur Rahman, Rachit Tamrakar

Department of Mechanical and Aerospace Engineering, Monash University, Clayton, Melbourne VIC, 3168

INTRODUCTION

A robust, unified, and scalable ground station and communication system forms the backbone of *Project Zenith* — *Monash High-Powered Rocketry's* (Monash HPR) pursuit of an innovative, competitive, reliable, and above all, safe rocket. The Unified Ground Station (UGS) is engineered to provide real-time control, continuous monitoring and data storage capability. It plays a critical role in managing the rocket's subsystems under dynamic and often unpredictable conditions. With a focus on clarity and operator support, UGS delivers actionable insights that enables informed, timely, and safe decision-making. Complementing the UGS, the *Intra-Rocket Interface System* (IRIS) serves as a light-weight, fault-tolerant communication protocol for on-rocket devices that fails loudly and clearly in the event of critical errors or communication breakdowns. Together, these systems are essential in safely operating several advanced Student Researched and Developed (SRAD) subsystems — including the hybrid propulsion system, primary deployment flight computer and Air Brakes — all of which require strict and reliable control to ensure *Project Zenith* can only transition between known and safe states.

THE UNIFIED GROUND STATION

For earlier projects within Monash HPR, each subsystem requiring real-time control or telemetry had its own dedicated control station, regardless of whether it needed only basic temperature data during launch or continuous monitoring throughout flight. As *Project Zenith* introduced increasingly complex systems with overlapping data requirements and wireless communication challenges, it became clear that this fragmented approach was no longer sustainable. To tackle this issue, the avionics team initiated a complete redesign, resulting in the creation of the Unified Ground Station (UGS).

The UGS was designed with reliability and safety in mind. A key design principle is *decoupled operation*, which allows subsystems to operate independently. This means that if a component fails, the failure can be isolated while the rest of the system is safed — giving operators the ability to respond, recover, or shut down in a controlled manner. To achieve this, UGS uses a modular architecture centered around *MQTT*, a lightweight publish/subscribe messaging protocol. This broker-based setup allows subsystems to connect as independent clients, enabling seamless fault isolation and system recovery. It also provides a scalable foundation where new subsystems can be added by introducing new clients, while maintaining a *single source of truth* for all critical data across the system.

To effectively manage the complexity of many simultaneously operating subsystems, UGS uses *Ignition*, an industrial, Commercial Off-The-Shelf (COTS) platform to deliver a clear, reliable, and highly customizable user interface. Ignition allows the team to quickly build web-based dashboards with tailored views for different operators, ensuring that each user sees the information most relevant to their role. Its unified development environment accelerates interface creation and simplifies ongoing updates, making it a practical and scalable choice for both current and future needs.

INTRA-ROCKET INTERFACE SYSTEM (IRIS)

While industrial COTS software offers enormous advantages — especially for a fast-paced student team like Monash HPR — not all problems have ready-made solutions. When existing tools fall short, custom solutions are developed in-house to meet specific needs. Inspired by standards like DroneCAN and Unified Diagnostic Services (UDS), the *Intra-Rocket Interface System* (IRIS) is a lightweight, asynchronous communication protocol developed in-house, specifically for embedded systems aboard rockets.

IRIS supports both peer-to-peer services and broadcasted messages. Services allow one node to directly request an operation from another, such as querying a sensor or initiating a reset, while messages handle system-wide coordination tasks like sequencing and data synchronization. Each service transaction is considered complete only when it either times out or the target device responds, including the possibility of a *Negative Response Code* (NRC) in case of failure. This explicit response mechanism ensures that faults are immediately visible, enabling quick operator intervention and reducing the risk of silent system failures.

The protocol's architecture emphasizes simplicity at the subsystem level. Each node is responsible only for basic verification and executing single-step commands such as sensor readings or valve actuation. More complex tasks such as command validation, sequencing, and decision-making are handled upstream, which is typically the UGS. This separation of responsibilities significantly reduces the likelihood of logic errors or unintended interactions, while enhancing system transparency and reliability.

To accommodate the varied communication needs within a rocket, such as multi-node coordination, long-distance wired links, and wireless interfaces, IRIS is designed to be agnostic to the physical layer. This flexibility promotes a unified system architecture, allowing seamless data exchange between subsystems while minimizing the need for additional hardware. As a result, IRIS strengthens *Project Zenith's* overall integration and streamlines communication throughout the vehicle.

APPLICATIONS

One of the most impactful applications enabled by this system architecture is the automatic fill process for Solaris MK II, Monash HPR's SRAD hybrid engine. This system automates a critical part of the pre-launch process — dynamically venting and filling the oxidizer tank to achieve a target propellant mass and temperature, based on real-time data from embedded sensors measuring pressure, mass, and temperature.

Coordinating fast, low-latency communication across three different embedded systems within the engine would traditionally be complex and error-prone. However, with IRIS, each embedded node transmits cyclic messages that continuously update the UGS with live telemetry. The UGS then issues service requests to adjust the onboard Active Vent valve in real time. Simultaneously, the UGS also controls the nitrous fill valve, even though the valve operates through a separate, off board system. This unified control strategy enables precise, closed-loop operation with minimal operator input.

On the User Interface (UI), the automated fill process is managed by simply setting a target mass and temperature, followed by a single button press. This drastically reduces operator workload and cognitive overhead, while maintaining a high level of system transparency and control.

Within *Project Zenith*, IRIS operates over a CAN bus network, supporting efficient half-duplex communication between all subsystems. A dedicated gateway bridges this network to the UGS using a serial RS422 link, simplifying overall wiring and enhancing scalability. This setup allows new devices to be integrated with minimal configuration or hardware changes. For example, it enabled the Air Brakes system to receive real-time temperature data without requiring additional dedicated wireless systems, highlighting the flexibility and efficiency of the unified communication framework.

CONCLUSIONS

The development of UGS and IRIS marks a pivotal step in Monash HPR's evolution toward a more integrated, scalable, and fault-tolerant launch infrastructure. By moving away from fragmented, ad-hoc systems, the team has created a robust foundation that enhances safety, simplifies subsystem integration, and supports real-time, closed-loop control across both ground and onboard systems. IRIS enables seamless, low-latency communication between embedded devices, while UGS consolidates control and monitoring into a unified, operator-friendly interface. Demonstrated in high-impact applications such as the automated fill process for Solaris Mk II, these systems reflect a broader shift toward deliberate systems engineering, laying the groundwork for continued innovation as the team pursues increasingly ambitious flight goals.

Candlebark: Hybrid Rocket Motor Simulation Suite for High-Power Student Rocket

Team 704 Project Technical Presentation to the 2025 IREC

Khoi Nguyen, Ethan Englund, Mitchell Galletly
The University of Sydney, Sydney, New South Wales, 2008, Australia

INTRODUCTION

Hybrid rockets are a popular choice in rocketry due to their technical simplicity and safety. Using liquid oxidizer and a solid fuel grain leads to simpler feed systems with denser solid fuel whilst retaining the ability to throttle and restart. Although simple in principle, modeling the performance of the engines is ever-evolving and very dependent on experimental results. USYD Rocketry Team has developed a modeling and simulation suite, Candlebark, that uses the latest models in two-phase flow, non-ideal non-adiabatic fluid dynamics, and hybrid rocket engine combustion combined into software package.

EXISTING MODELS

Current existing solutions are restricted in many ways, including but not limited to non-ideal non-adiabatic behavior in blowdown, which does not capture transient tank behavior accurately. Furthermore, most existing solution implements a single-phase compressible flow model for simplicity; however, it does not effectively capture cavitation and compressible flow effects through the injector orifice, which will cause inaccuracies in high pressure-drop engine configurations. Finally, Candlebark can also support fuel grain geometry modeling for any arbitrary polygon and multiport configuration.

SIMULATION OVERVIEW

Candlebark discretizes the time domain of the simulation and solves for the state derivative of the engine every timestep. The initial state of the tank is found by determining the saturation pressure of nitrous oxide based on the fluid temperature via the thermofluids library CoolProp. From this, the molar fractions of liquid and vapor can be determined. Given the known upstream state of the oxidizer and the downstream combustion pressure, Candlebark calculates the mass flow rate through the injector orifice. From the drain of the oxidizer, the new tank temperature and pressure can be deduced by utilizing the Peng-Robinson Equation of State. The fuel mass flow rate can then be calculated via the spatially averaged Marxman regression model, which has ballistic coefficients derived from an extensive propellant characterization campaign. Based on the regression rate, the new fuel grain geometry is calculated using a Minkowski Sum approach. The flame combustion properties can be found by utilizing NASA's Chemical Equilibrium Analysis. Finally, an isentropic model is used to calculate the nozzle's exit velocity and mass flow rate to determine the thrust force per timestep.

APPLICATION TO PARDALOTE

Candlebark proved to be a foundational tool in the preliminary design cycle of Pardalote, USYD Rocketry Team's first ever high powered hybrid rocket. Candlebark enabled a large parametric sweep of components critical to design such as tank volume, injector area, fuel grain geometry and nozzle sizing. These simulations were executed concurrently to each other, allowing faster processing times. Beyond engine sizing efforts, Candlebark was also integrated as a mission design tool, cooperating with the ground control electronics to provide live apogee updates depending on the loaded propellant mass and saturation pressure. It can also be used to size a preliminary ballast to help Pardalote hone in on the 10,000 feet target apogee.

CONCLUSIONS

The development of Candlebark significantly enhanced the flexibility of the Pardalote propulsion design process, allowing the team to move beyond empirical relations and simple modelling and simulation. In the future, further improvements are planned to enhance modeling resolution and realism. In particular, it expands the multiport regression model to focus on flow effects on different ports. Furthermore more research should be conducted into modelling the axial variations in regression.

Safety Critical Software Design for a Hybrid Rocket

Team 706 Project Technical Presentation to the 2025 IREC

Shivam Desai and Christopher Chan
University of Calgary, Calgary, Alberta, T2N 1N4, Canada

INTRODUCTION

Unlike solid rockets, hybrid propulsion rockets require far superior monitoring and control systems. Hybrid systems demand precise coordination of fill operations, ignition control, and continuous sensor feedback, making the operator's role more active, and the software stack more critical. These systems demand extensive software control and allow for more active decision making throughout the launch procedures. While we try to mitigate much risk through operations procedures there are increased risks with this electronic control: incorrect operator inputs, power failures, communication faults, and software crashes. Building safety critical software would mean being able to detect and respond to control and anomaly events in a timely manner.

Real Time Operating System

Our system had to be designed to handle sensor data collection and storage, communication of this sensor data and react to real time commands like valve actuations. These conflicting demands necessitate a reliable and deterministic operating system architecture, where tasks are scheduled based on priority and timing requirements rather than brute force or manual logic.

A Real-Time Operating System (RTOS) is an operating system designed to perform operations in a precise amount of time, respecting well-defined deadlines. In safety-critical environments, timing is often more important than raw processing power and violating these timing deadlines could mean that the system could severely injure other systems, people and objects.¹

To meet these needs, we have built our own operating system around FreeRTOS, an open-source Real-Time Operating System kernel. The kernel facilitates the interactions between hardware and software for the computing system. More importantly for us, the kernel facilitates concurrent execution of multiple tasks, including sensor data collection, actuator control, debugging, and communication. FreeRTOS provides task management, inter-process communication, and real-time scheduling capabilities, all within a lightweight footprint suitable for embedded systems.⁴

Some tasks in our system are far more time-critical than others. For instance, if a command to actuate a vent or Main Engine Valve (MEV) arrives while logging sensor data, the actuation must take priority. This is where preemption in Real-Time Operating Systems (RTOS) is vital. Preemptive scheduling lets high priority tasks interrupt lower priority ones, ensuring time sensitive actions are handled immediately and making the system safer to operate.

SOAR OS

In software it is critical to have a consistent way to communicate between modules written by different developers. The development of a wrapper around commonly used software primitives and using a consistent pattern for inter-task communication enables smoother communication along with supporting debugging purposes.

Leveraging C++ polymorphism, we created base classes that define the generic structure and behavior of each thread (or task in FreeRTOS). Each task is a singleton, ensuring only one instance runs at a time, which improves predictability and prevents duplication of critical operations. Tasks extend from this base to serve specific roles while maintaining core behaviors like safe initialization, standardized execution, and a shared communication protocol. This object-oriented design promotes modularity and maintainability.

We enforce standardized inter-task communication through "command" objects that encapsulate both instructions and data payloads. Each task has a dedicated queue, and other tasks interact with it by enqueueing these commands. This ensures thread-safe, isolated data handling and avoids concurrency issues such as race conditions.

This operating system design allows us to run concurrent processes and has become the backbone to extend system functionality safely and deterministically.

Rocket State Machine

At its core, a state machine is a conceptual model used to describe a system in terms of its states, the transitions between these states, and the actions that result from these transitions. It's a way of breaking down complex behaviors into manageable chunks, making it easier to design, understand, and debug systems.³

One way we mitigate operational risk is through the development of our own Rocket State Machine (RSM). As stated above, the state machine allows us to define the different states in which our rocket will be operated and define transition behavior when entering and exiting a state, ensuring that the system behavior is always deterministic. In our RSM we have states such as Test, Prelaunch, Fill, Arm and Abort. Using our state machine, we define this behavior for different cases such as, opening both the vent and drain valves when entering the Abort state. Hence, if there is any emergency or an automated abort is triggered then the system will safely depressurize the rocket's pressure vessel, making it safe to approach.

The state machine also constrains operations within every state. For example, we block any commands attempting to open the MEV during the Fill state or setting off the ignitors when in pre-launch state. These rules enforced by our RSM, help reduce operator error as well as the risk to on-site system operators.

Finally, we also have a feature called "state recovery" implemented in our software systems. With most devices if there is any power failures or intermittent power cuts, they will completely reset. In the case of our hybrid rocket, this can become very dangerous. If we had intermittent power during our propellant fill or halfway through our flight, then the vent and drain valve would open up to their default state. Utilizing our RSM and onboard flash chip, we store our latest system state into non-volatile memory. This allows us to check the previous system state on power up and restore it as needed. Having implemented and tested this we were able to achieve a recovery time of 50ms, greatly reducing the impact of an unprecedented power failure.

Communication Systems

A robust communications system is critical for safe operation of a remotely operated system. Simultaneously, it is critical to have a low overhead and latency to enable faster telemetry rates and reduce risk for time critical operations such as main engine valve actuation from a remote sensor source. To enable consistent communication between all nodes in our system on the rocket and ground systems, we utilized Protocol Buffers. Protocol buffers allow you to define how you want your data to be structured once, then you can use special generated source code to easily write and read your structured data to and from a variety of data streams and using a variety of languages.⁵ This central data format minimizes the risk of inconsistencies that could compromise sensor data or control commands. All messages are wrapped using COBS (Consistent Overhead Byte Stuffing) encoding to reliably frame data by removing 0x00 bytes from the stream.

During systems operation there is also the concern that data being sent and received may be corrupted and misinform the system operator potentially leading to adverse side effects. As a mitigation, each message also includes a 2-byte CRC-XMODEM checksum, verifying the integrity of the data being sent and received, allowing us to discard corrupted data packets. Additionally, the communication protocol includes explicit acknowledgments (ACK) and negative acknowledgments (NAK) for control commands. This creates a closed-loop feedback system where commands are only considered successful once confirmation is received from the rocket's electronics.

No system is immune to failures. To mitigate the effect of communication system failures, RTOS freezes, or other issues, we implemented a watchdog timer (heartbeat). This countdown resets regularly via backend messages from the control interface. If the system fails to receive this signal, the RSM forces the rocket into an Abort state. This automatic fail-safe prevents the rocket from remaining in a dangerous condition if control is lost.

RESULTS, CONCLUSIONS, AND FOLLOW-ON WORK

The implementation of a RTOS, thread safe program execution, rocket state machine and finally communication systems has constructed a robust, real-time architecture that can operate reliably in the face of uncertainty. In past tests we have witnessed features such as the heartbeat in action when losing telemetry with the rocket and the rocket state machine successfully prevented accidental operator risks, rejecting unintentional commands when necessary. Having this foundation we now look forward to implementing more features to continue to improve system performance and safety. For example, our switch to DMA (Direct Memory Access) protocols, will allow us to offload process overhead from data logging systems to a separate controller, freeing up the CPU to handle more flight critical requests and allowing for uninterrupted data logging. In the high stakes environment of rocketry, this software system ensures that we can avoid catastrophic system failures, while still providing us with a large amount of control.

REFERENCES

¹D. Mamone, A. Bosio, A. Savino, S. Hamdioui and M. Rebaudengo, "On the analysis of real-time operating system reliability in embedded systems", *Proc. IEEE Int. Symp. Defect Fault Tolerance VLSI Nanotechnol. Syst. (DFT)*, pp. 1-6, Oct. 2020.

²Luna, Ryan, and Sheikh Ariful Islam. "Security and reliability of safety-critical rtos." *SN Computer Science* 2.5 (2021): 356.

³Wagner, F., Schmuki, R., Wagner, T., & Wolstenholme, P. (2006). *Modeling software with finite state machines: a practical approach*. Auerbach Publications.

⁴*FreeRTOSM - FreeRTOSM*. (n.d.). <https://www.freertos.org/>

⁵*Protocol buffers*. (n.d.). <https://protobuf.dev/>