

Rocketsmith: Agentic Additive Manufacturing of High-Powered Rockets

Peter Pak,^{†,‡} Jesse Barkley,^{†,‡} Rumi Loghmani,^{†,‡} Derek Baich,[‡] Ananya Pamal,[†]
and Amir Barati Farimani^{*,†,¶}

[†] *Department of Mechanical Engineering, Carnegie Mellon University, Pittsburgh, PA, USA*

[‡] *Tripoli Rocketry Association, Pittsburgh, PA, USA*

[¶] *Machine Learning Department, Carnegie Mellon University, Pittsburgh, PA, USA*

E-mail: barati@cmu.edu

Abstract

RocketSmith is an agentic system which intelligently automates the DFAM process for the development of high powered rockets suitable for launch. The system utilizes a large language model to orchestrate the execution of software tools to validate design characteristics such as flight stability and generate the parametric design components for the rocket assembly. A collection of subagents and skills enable optimization workflows of flight parameters via iteration in both zero-shot and human-in-the-loop workflows. With this system, four distinct high power rockets with various motor and assembly configurations were developed utilizing the unique design capabilities of additive manufacturing. These assembly components were fabricated using various FDM printers, manually evaluated for flight readiness, and flight tested at a launch event. From these tests, all rockets achieved a stable launch and two of the four rockets were successfully recovered in reflyable condition. The altimeter data validated that the rockets achieved an altitude 80% of the expected apogee predicted by the agentic system, establishing consistency between simulation and experimentation.

1 Introduction

Additive Manufacturing (AM) enables design and fabrication capabilities outside the realm of conventional manufacturing techniques such as iterative design, rapid prototyping, and part consolidation.¹ These considerations make the utilization of AM attractive to development constrained and cost elastic industries, specifically aerospace.² Within the development of an assembly there are a number of factors outside the scope of manufacturing that have an effect on its production application. High powered rocketry presents an environment where these multifaceted design and manufacturing are encountered and where additive manufacturing presents a unique solution to part consolidation and flight optimization. This work explores the development of an agentic system, RocketSmith (Figure 1), capable of streamlining the design and manufacturing process for the development and flight testing of Class 2 high powered rockets.³

Development of high powered rockets is often an iterative process as design specifications can change due to factors realized in manufacturing, part availability, and other unforeseen events. This contributes to an iterative loop where variables such as stability are recalculated with component values obtained after manufacturing, informing design and manufacturing decisions for the other components within the assembly. Within the iteration loop there is noticeable friction between the various software tools for flight simulation, design, and manufacturing.^{8,9} The flight timeline of a high powered rocket is composed of a series of events summarized as: motor ignition, motor burnout, flight apogee, recovery deployment, and retrieval. The propellant from motor ignition provides the necessary thrust to lift the rocket upward until the motor burnout event.^{10,11} The momentum following the motor burnout event continues to carry the rocket upward until it reaches its apogee where the deployment event occurs.¹⁰ At the apogee, velocity is minimal and for single deployment systems the main parachute deployment event occurs. However, for dual deployment systems the drogue chute deployment occurs at the apogee for a controlled descent before the main parachute deployment event is executed.¹⁰ Following this sequence of events the final recovery event

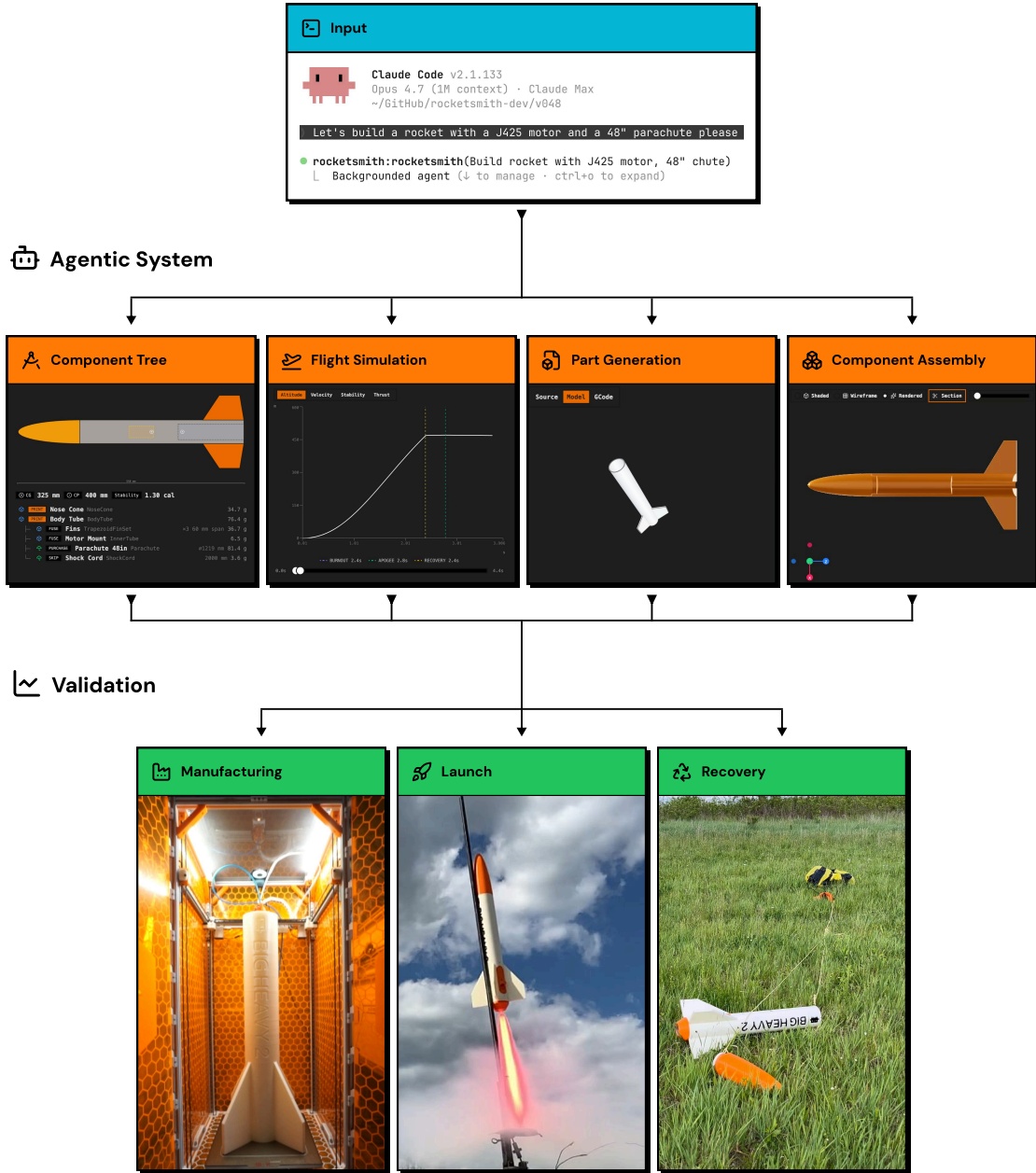


Figure 1: RocketSmith utilized as a Claude Code⁴ plugin enables the efficient development of high powered rockets. The agentic system is capable of designing an OpenRocket⁵ based component tree with provided user constraints such as specific motors and flight characteristics. Prescribed dimensions from the component tree are utilized to generate parametric models of the rocket airframe using CADSmith.⁶ More precise weight calculations obtained with PrusaSlicer⁷ improve center of gravity estimations and allow for iterative optimization of the rocket’s stability before manufacturing. Each part is additively manufactured via FDM where afterwards the components are assembled, flight tested, and successfully recovered.

involves the retrieval of the rocket from its landing site.¹⁰

Agentic system based approaches to problem solving have displayed considerable capability in technical fields for applications such as molecular design,^{12,13} robotics,^{14,15} additive manufacturing,^{16–18} software development,^{19,20} materials science,^{21,22} and mechanical design.²³ The parametric knowledge embedded within the agentic system’s Large Language Model (LLM) provides the model with not only an enhanced dataset of training data to extrapolate from but also enables complex reasoning capability necessary for domain specific tasks.^{24,25} In addition, these LLMs provide the foundational reasoning and tool calling abilities allowing interactions with the surrounding environment.^{12,18,23} These set of capabilities make the use of agentic systems attractive to the iterative development cycle of high powered rockets.

This work introduces RocketSmith, an agentic system composed of subagents, skills, and tool calling abilities developed to the standard of the Model Context Protocol (MCP) and released as a plugin for agent harnesses such as Claude Code. This system is capable of orchestrating the necessary flight simulations, designing the appropriate CAD files, and generating manufacturing files of a high powered rocket with user provided constraints and specifications. The schematics produced by RocketSmith are then additively manufactured via Fused Deposition Modeling (FDM) and assembled for evaluation through a series of flight tests at a following launch event (Figure 2). Quantitative altimeter data collected from a subset of flight tests along with qualitative metrics are used to evaluate the general performance of the agentic system. The code is offered in the form of a plugin / extension for Claude Code and is available at <https://github.com/ppak10/RocketSmith>.

2 Related Works

LLM-3D print by Jadhav et al.¹⁶ investigates the application of a large language model (variants of ChatGPT 4 by OpenAI) enabled agentic system for optimizing fused deposition



Figure 2: (Left to Right) High Power 1 launched by Pak with an AeroTech H100W motor, High Power 2 launched by Loghmani with AeroTech H219T motor, High Power 3 launched by Barkley also with AeroTech H219T, and High Power 4 launched by Pak with an AeroTech J425R motor.

modeling printing parameters. By leveraging the vision capabilities of ChatGPT 4, optical images taken of the build layer are evaluated by the LLM for any potential defects and the necessary actions are executed to address these issues. The agentic system utilizes the LangChain library as its primary framework where in which the supervisor agent orchestrates tasks to its subagents for tasks such as planning, information extraction, and solution execution.¹⁶ These actions significantly improved the print quality of tested parts, addressing potential defects in near real-time, further validated by compression tests with increased peak load sustained by LLM-3D print fabricated parts compared to parts build without the assistance of the agentic system.¹⁶

RocketBench by Simonds et al.²⁶ is a benchmarking tool which connects LLMs to high-fidelity trajectory simulation of RocketPy²⁷ to evaluate each LLM’s capability to design rockets. Models specify designs through a structured JSON interface and are scored on a composite reward that combines altitude accuracy, structural integrity, horizontal drift, cost efficiency, and landing safety. The authors find that while frontier LLMs demonstrate strong baseline engineering intuition, these models consistently plateau below human expert

performance during iterative refinement, exposing the fundamental limitation that standard LLMs struggle to translate simulation feedback into meaningful design updates.²⁶ These findings motivate several architectural decisions implemented in RocketSmith, where rather than relying on the LLM to iteratively optimize design parameters directly, delegating these numerically precise tasks to tools would result in accurate responses. These include tasks such as stability calculation, trajectory simulation, slicing, and mass estimation. This addresses the core failure mode identified in RocketBench where instead of asking the LLM to reason about simulation feedback and update parameters itself, the design process is structured so that each stage produces verifiable outputs for the next stage to consume deterministically.

Early text-to-CAD approaches such as CAD-LLM²⁸ and LLM4CAD²⁹ framed CAD generation as a sequence modeling problem, prompting models to produce CAD command sequences from natural language. These methods demonstrated basic geometric reasoning but struggled with dimensional precision, as parametric errors compound through a modeling sequence without any external correction signal. More recent systems have moved toward the Agent-Aided Design paradigm, placing an agent in a feedback loop that iteratively generates, compiles, and visually inspects geometry.⁶ CADSmith⁶ provides a multi-agent text-to-CAD solution that generates CadQuery³⁰ scripts from natural-language prompts and refines them through nested correction loops. It is composed of an inner loop that catches Python and OpenCASCADE³¹ execution errors and an outer loop validates generated geometry against exact measurements. These measurements (bounding-box dimensions, volume, solid validity) are extracted from the OpenCASCADE kernel, supplemented by an independent vision-language model judge. Retrieval-augmented generation over the CadQuery API documentation supplies type-correct code patterns without fine-tuning. On a 100-prompt benchmark spanning three difficulty tiers, CADSmith raised execution rate from 95% to 100% and reduced mean chamfer distance from 28.37 to 0.74 against a zero-shot baseline, demonstrating that programmatic geometric validation resolves a class of dimensional errors that visual-only feedback cannot.

3 Background

3.1 High Power Rocketry

High power rocketry (Class 2) is an intermediate category of rocketry with a total impulse ceiling of 40,960 N · s.³ Access to respective impulse ranges (Appendix A) are granted through certifications offered through organizations such as Tripoli Rocketry Association (TRA) and National Association of Rocketry (NAR), both work with the Federal Aviation Administration (FAA) who administers flight waivers. This class of rockets is most accessible to adult hobbyists where motors are permitted for purchase with the appropriate certifications. Advanced high power rocketry (Class 3) covers rockets beyond 40,960 N · s and up to 889,600 N · s and requires additional permission from the respective governing bodies.³

3.1.1 Stability Calculation

Stability calculation is a critical component during the design process. Rockets with low stability are more likely to tumble during launch, however an excessive stability value is prone to directional changes from environmental factors such as crosswind in a phenomenon called weathercocking.¹⁰ For these reasons, a stability value between 1.00 cal and 1.25 cal is desired. Stability is determined with three main variables: Center of Pressure (CP), Center of Gravity (CG), and body tube diameter.¹⁰ Locations for the center of gravity x_{CG} and center of pressure x_{CP} are measured from the nose tip to the tail and distance between the two is divided by the body tube diameter d to provide the stability value in the unit of calibers (Eq. 1).¹⁰

$$Stability = \frac{x_{CG} - x_{CP}}{d}, \quad x_{CG} = \frac{\sum_i m_i x_i}{\sum_i m_i}, \quad x_{CP} = \frac{\sum_i C_{N_i} x_i}{\sum_i C_{N_i}} \quad (1)$$

The center of gravity is weighted average of the individual mass components m_i and their respective distance from the nose tip x_i .¹⁰ The center of pressure utilizes the Barrowman³² equation which considers the individual normal force coefficients C_{N_i} of each respective

component to calculate the weighted average.¹⁰ During the development of the rocket, the calculated stability often fluctuates on the basis of the recorded mass of each manufactured component. In addition to the structural components of the rocket, consideration needs to be given to the motor's change in mass during launch as this often shifts the center of gravity towards the nose tip.

3.1.2 Component Design, Manufacturing, and Assembly

The design and manufacturing of the various rocket components are key phases foundational to successful launch and recovery events.¹⁰ These phases take into consideration the various events that occur during and after the launch event such as the placement of rail buttons along the airframe and the internal pressure caused by recovery deployment charges. Thoughtful consideration during this design and manufacturing phase is critical to the successful recovery of the rocket after launch.

Weight is often the primary constraint within the design process; however, other considerations such as accessibility, manufacturing, and material are also non-trivial factors.^{10,33} Material selection is often settled early in the design phase as it establishes initial constraints of weight, manufacturing, and size.³³ These pertain to parts such as body tubes, fins, nose cone, and other internal components with common material candidates of cardboard, wood, fiberglass, carbon fiber, and metal alloys.^{10,33} Decisions regarding material selection are often tied with considerations towards manufacturing as each material presents its own set of manufacturing challenges.

The assembly of manufactured components is often the final phase before launch preparation. At this phase, the manufacturing tolerance of components is assessed as each part is required to fit with their respective mates to ensure correct flight behavior. Parts that are out of specified tolerances need to be adjusted through either subtractive or additive modifications to fit correctly within the assembly. In addition, flight parameters such as the actual center of gravity are recorded from the rocket assembly, providing real values to

adjust flight simulations.

3.2 Agentic Systems

An agentic system is a platform where an LLM is able to autonomously complete multi-step goals by means of context, reasoning, tool calls, and optimization.^{12,16,18} Agentic systems are useful for complex tasks as the system is capable of operating beyond the constraints of a single input prompt and the domains of the training data.^{24,34} These systems are primarily composed of three main components: LLM enabled reasoning and orchestration, tool calls for precise and accurate responses, and an agent harnesses which provides the runtime for the LLM and tools. Together this combination of components provide the ability to ingest natural language task descriptions, decompose and execute subtasks, and validate the results.

3.2.1 Large Language Models

Large Language Models (LLMs) are transformer-based neural networks trained on trillions of text tokens through next-token prediction.²⁵ Transformers use self-attention to calculate weighted relationships between tokens of a sequence in parallel. Next-token prediction is the process of generating a probability distribution of vocabulary given a sequence of tokens. The model parameters are updated to maximize the likelihood of the “true” next token.²⁵ Tokens are sub-word units that are made from byte-pair encoding.³⁵ After pretraining, these models undergo post-training consisting of supervised fine-tuning on instruction-response pairs and subsequent reinforcement learning from human feedback or direct preference optimization to create instruction-following generations.^{36,37} The generation process itself is autoregressive, which means that at inference time the model samples one token at a time from the output distribution and appends it to the input. This process repeats until the end of a sequence or a predefined maximum length.

The context limit is the maximum number of tokens that an LLM can perform attention upon.²⁵ This parameter is a key architectural factor during the development of the model

and fixed during training. Modern frontier models have context windows ranging from 128k beyond 1M tokens. Compute scales quadratically with sequence length $O(n^2)$ in standard attention, resulting in increasing cost and latency for longer prompts.²⁵ Liu et al.³⁸ showed that accuracy on retrieval-style tasks degrades to near or below performance on shorter contexts when the relevant information is in the middle of the window. Within agent harnesses handling large amounts of contextual information, subagents and tool calling can be used to ensure consistent responses and behavior.

3.2.2 Tool Calling

Tool calling allows for LLMs to interact with their external environment with predetermined functions.^{24,34,39} Early examples of this include WebGPT³⁴ which utilized a simple set of tools to navigate and collect information through various websites through the guidance of an LLM. Within the process, the LLM is given a context schema that provides details on how each tool should be called and in what context it should be called upon. Since these tools are often snippets of pregenerated code, this allow LLMs to produce deterministic outputs by invoking external code. The LLM itself is not capable of executing the code, rather it relies on the higher order function performing LLM inferences to execute functions based on parsing the generated response.²⁴ The tool call then generates a response that is then returned to the LLM in the form of additional context to the conversation.^{12,16,18} Here the LLM acts as an orchestrator for tools calls, utilizing its reasoning and context to make informed decisions.²⁴ Within tool calling there are failure modes such as schema mismatches and semantic errors which can be avoided through use of a standard protocol such as the Model Context Protocol (MCP).

3.2.3 Agent Harnesses

An agent harness provides the runtime for LLM inference and other functionalities such as tool calls and user interaction.^{24,40} The harness is responsible for managing conversation

history, executing tool calls, providing system access, context window management (summarizing or removing previous turns), and termination conditions.^{4,41,42} A common workflow involves the agent receiving user instructions, passing these to the LLM, executing necessary tool calls, appending outputs to the conversation, and repeating until a termination condition is met. Common terminal based agent harnesses include Claude Code, OpenCode, Antigravity CLI, and Codex CLI which provide a terminal based user interface for interacting directly with the LLM.

Within agent harnesses, further functionalities include skills, subagents, and hooks. These features allow for a level of customization enabling further control of the agentic system. A subagent is a child agent instance that maintains its own context window, system prompt, and tool subset, when instantiated by the parent agent.⁴³ Subagents are often invoked for sub-tasks that work within their own context window and prevents the parent agent’s context window from pollution during subagent task exploration.⁴³ A summary of a subagent’s final response is then appended to the conversation and used in subsequent inferences.

4 Methodology

4.1 Software Tools

The core of RocketSmith utilizes three main software tools: OpenRocket,⁵ `build123d`,⁴⁴ and PrusaSlicer.⁷ OpenRocket provides the design and simulation tools to generate rocket blueprints, `build123d` offers the capability to generate parameteric **STEP** files of the declared design dimensions, and PrusaSlicer calculates more concrete weight estimations and tool path files for the generated components. With the data obtained during this pipeline, adjustments are made within the OpenRocket platform and the updates are made using the downstream software tools. The agentic system primarily relies on these three main software tools to enable the intelligent automation of component design and manufacturing.

4.1.1 OpenRocket

OpenRocket⁵ is a design and simulation tool commonly used for the development of high powered rockets. This platform provides a database of commonly used rocket components such as solid motors, body tubes, parachutes, nose cones, and other launch accessories useful for designing the build of a rocket.⁵ These components along with their respective properties such as mass, dimensions, and material are utilized during the design procedure and provide valuable insight into flight metrics such as stability and apogee. The design is utilized for the basis of flight simulations that predict the altitude, velocity, acceleration and other flight characteristics using variables such as wind conditions, recovery deployment, and launch rail length. These flight simulations are valuable as they provide a general estimation of the course of the flight test before any of the components are built.

As components are manufactured, various dimensional and weight estimations often deviate slightly from those prescribed in OpenRocket.⁵ These initial values can be overridden by actual values to utilize accurate variables during the generation of flight simulations. Other factors such as multiple stages or recovery device deployments can be adjusted on the software platform for further control in more complex configurations. During development, OpenRocket⁵ is often used as the design blueprint for any CAD, manufacturing, or assembly task.

RocketSmith uses OpenRocket⁵ as the primary source of truth during the course of rocket development (Figure 3). All dimensional, weight, and assembly configurations are first implemented within OpenRocket⁵ before being sent to any downstream software tools. Integration of the OpenRocket⁵ software into the agentic system is enabled using the OpenRocket Helper⁴⁵ package which provides Python bindings to the Java based OpenRocket⁵ platform. This allows the agentic system to utilize the database of existing components, create and modify components, perform overrides for mass and dimensions, and run flight simulations. The designs and simulations built using OpenRocket⁵ makes this software foundational to the reliable generation of flight capable rocket designs by the RocketSmith agentic system.

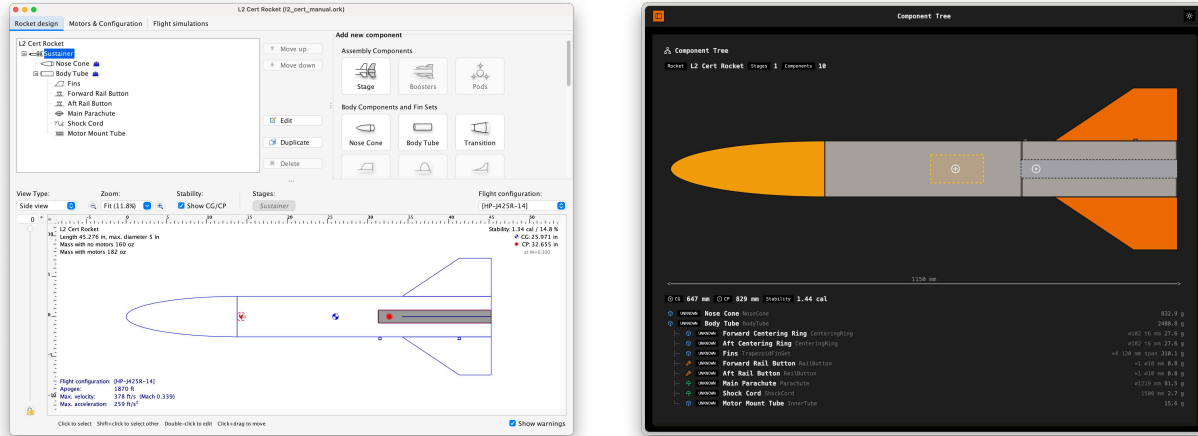


Figure 3: (Left) OpenRocket⁵ provides the foundational toolkit for generating rocket designs and running flight simulations. (Right) RocketSmith establishes a component tree to use for the generation of STEP file components and design related skills and subagents.

4.1.2 build123d

`build123d`⁴⁴ is a parametric CAD Python library that wraps the OpenCASCADE³¹ geometric kernel and uses context-manager and operator-overload conventions to define solids imperatively. This package is a successor to `CADquery`³⁰ and shares the same kernel, but exposes a more Pythonic authoring surface that maps cleanly onto the linear, top-to-bottom structure an LLM tends to produce, with named intermediate state at each step. Code-based formulation is adopted over generative-3D approaches (diffusion, point-cloud or mesh transformers) for four reasons specific to agentic authoring. Firstmost, the script is deterministic and parametric: identical inputs yield identical geometry, and the LLM can target a specific feature for adjustment (changing a single fillet radius or shifting one fin chord) without regenerating the entire shape, in contrast to diffusion approaches that must re-sample the whole solid for any local edit. Second, the output is a STEP file whose exact bounding box, volume, and mass (via material density) can be extracted programmatically and checked against the manifest’s expected values, making closed-loop verification possible without lossy visual proxies. Third, STEP is the industry exchange format, and flows directly into the slicer (PrusaSlicer) and any downstream finite-element or CAM tooling without

the mesh-cleanup steps that generative outputs typically require. Fourth, `build123d` fails surface as Python exceptions carrying a geometric reason, so the agent can read the traceback and adapt rather than producing silently invalid geometry.

Within RocketSmith, `build123d` is invoked once stability and flight readiness has been verified in the OpenRocket design and simulation phase. The CAD agent receives the finalized part list with each component’s dimensions and authors a `build123d` script for every printable part. As the scripts execute and STEP files are written, the corresponding parts appear progressively in the project GUI (Figure 4), allowing the user to observe build progress and issue iterative adjustments to the agent (for example, requesting a thicker wall, a longer shoulder, or a re-positioned vent). When all parts are complete, the GUI presents an assembly view rendering every `build123d` part in its correct relative orientation. The user can iterate back and forth with the agent on fine adjustments or alterations until satisfied, after which the final STEP files are exported for slicing and additive manufacturing.

4.1.3 PrusaSlicer

PrusaSlicer⁷ is a configurable CAD model slicer software forked from the Slic3r⁴⁶ project for Fused Deposition Modeling (FDM) tool paths. The slicer provides configurable parameters for three main sections: print settings, filaments, and printers. Configurations within print settings determine build specific settings regarding layer height, tool speed, infill, supports, and other print related settings. Material specific settings can be assigned in the filament section allowing for adjustment of bed temperature, hotend temperature, and various cooling parameters. The printers section provides machine specific specifications that need consideration during the slicing process such as dimensional limitations and number of available extruders. Within RocketSmith, PrusaSlicer (Figure 5) allows for a more precise weight estimation of the generated STEP file component through slicing with the expected material, print, and printer configurations. Once sliced, the generated tool paths are uploaded to the respective printer for manufacturing where once fabricated are weighed to obtain actual

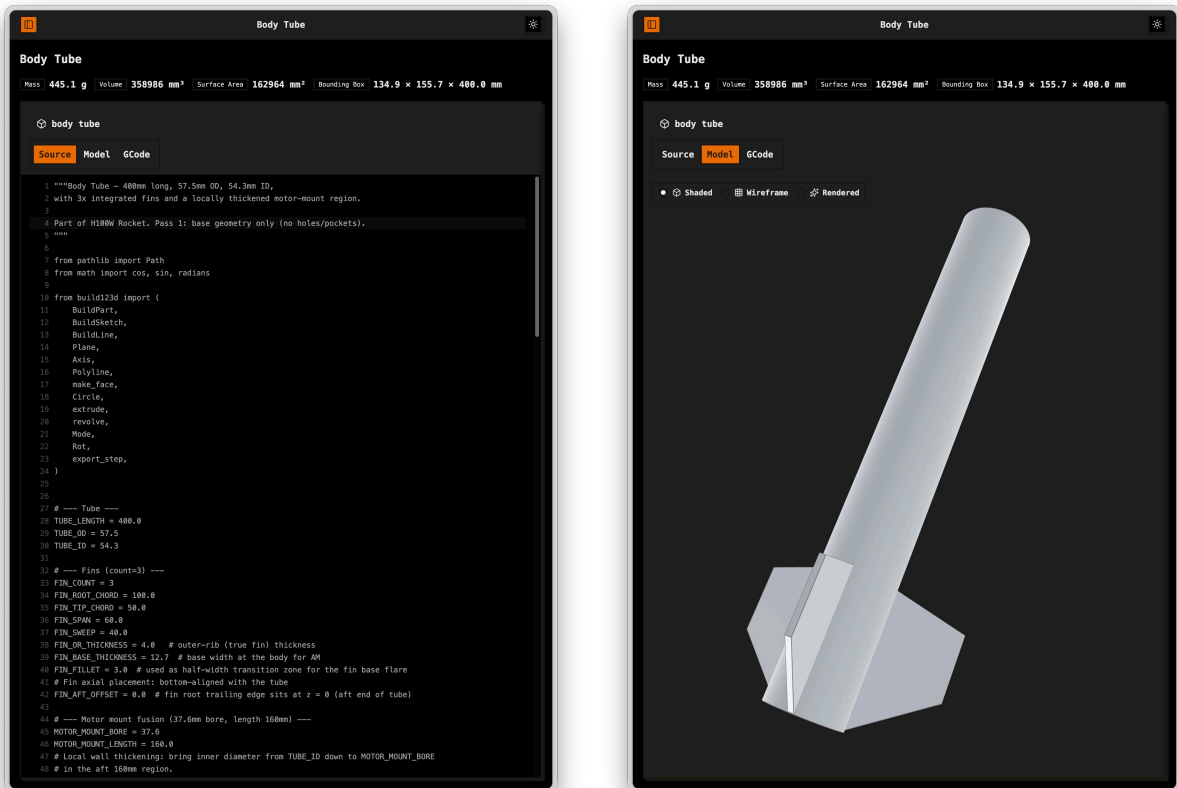


Figure 4: (Left) `build123d` is the primary framework to use for generating the parametric part configurations written in Python. (Right) STEP file is generated from running Python snippet and visualization allows for easy part adjustment.

weight overrides.

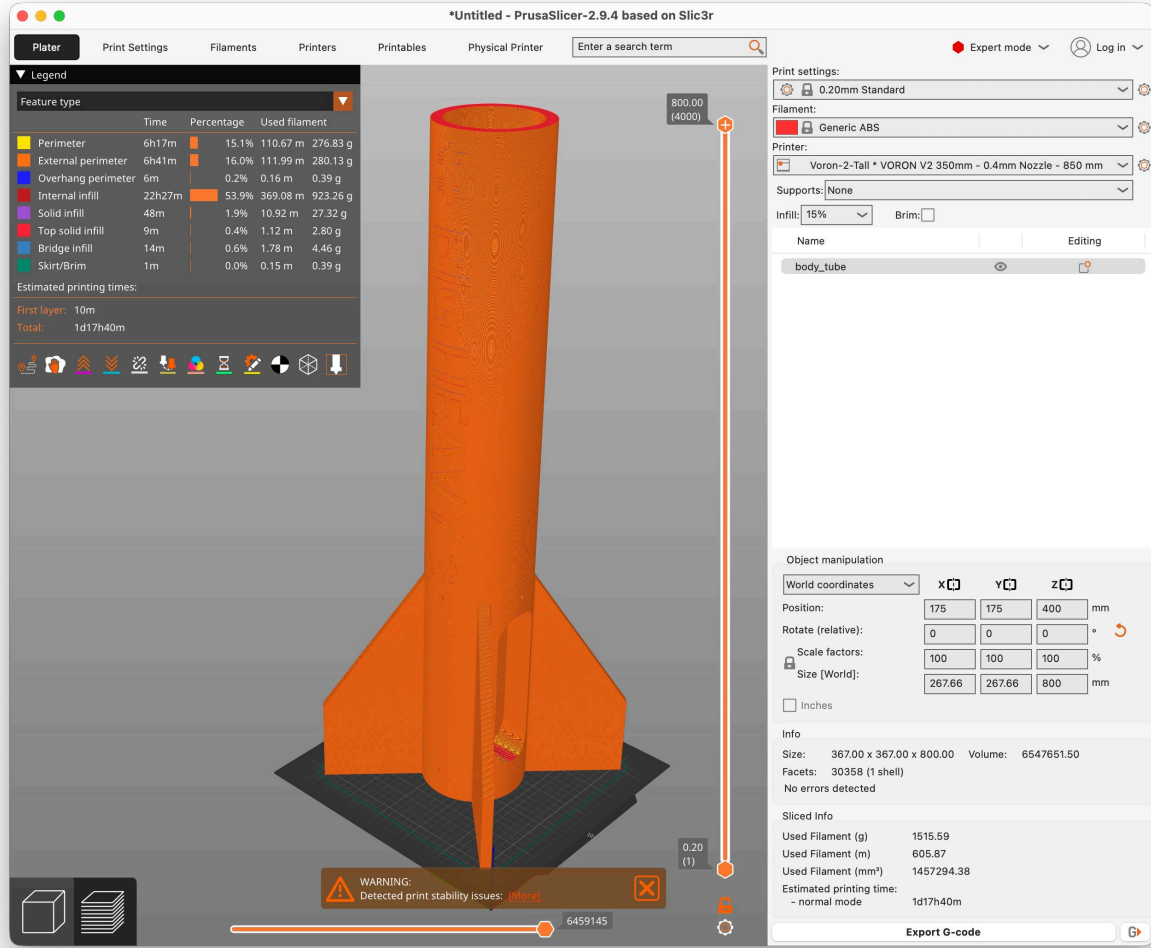


Figure 5: Weight estimation for generated STEP files for various components are obtained using Prusaslicer⁷ using configurations for the expected material.

4.2 Agentic System

The development of high powered rockets is augmented with the usage of an agentic system which is capable of operating in zero-shot^{47,48} and human-in-the-loop⁴⁹ conditions. Under both operating conditions the system provides a graphical user interface to monitor and evaluate the outputs including CAD models, flight simulations, and component trees. The use of subagents allows for the compartmentalization of specialized context specific to operations

concerning the OpenRocket, PrusaSlicer, CADSmith,⁶ and other RocketSmith functionality. Skills provide a concrete workflow for tool usage regarding specific behaviors that are common throughout the duration of the system.

4.2.1 Graphical User Interface (GUI)

The Graphical User Interface (GUI) provides a medium to observe the actions of the agentic system during the rocket development process, especially useful in design related tasks regarding assembly and modeling. The main dashboard of the GUI enables a holistic view into the various functions of the RocketSmith agentic system, highlighting the current active card the agent is concerned with (Figure 6). These cards include specific aspects of the agentic system including the rocket component tree, flight simulation results, CADSmith⁶ scripts and models, and assembly configurations. At its core, the GUI simply a “readonly” visualization of the current state of the agentic system as the primary method of executing actions is through the CLI based agent harness of Claude Code.⁴

4.2.2 Subagents

The agentic system is composed of 6 individual subagents: `rocketsmith`, `cadsmith`, `gui`, `manufacturing`, `openrocket`, and `prusaslicer`. Each individual subagent is responsible for a limited scope of the agentic system and enables efficient utilization of context. The `rocketsmith` subagent is responsible for the overall function of the agentic system denoting the project file structure layout, declaring subagent scopes, and applying various soft guardrails. Subagents for `openrocket`, `cadsmith`, and `prusaslicer` provide additional information regarding how to best utilize each respective software platform’s API and when it is appropriate to do so. The `manufacturing` subagent manages the hand off steps between the OpenRocket⁵ and CADSmith⁶ software programs evaluating potential Design for Additive Manufacturing (DFAM) and Design for Manufacturing (DFM) adjustments. The `gui` subagent manages the visualizations shown to the user, navigating between different pages

```

ppak@ms-pro-macbook-pro-8 v046 % claude --permission-mode bypassPermissions --plugin-dir ../../RocketSmith
Claude Code v2.1.133
Opus 4.7 (1M context) · Claude Max
~/GitHub/rocketsmith-dev/v046

) Let's build a rocket with a h100 motor and a 48" parachute

>> bypass permissions on (shift+tab to cycle)
Image in clipboard · ctrl+v to paste

```

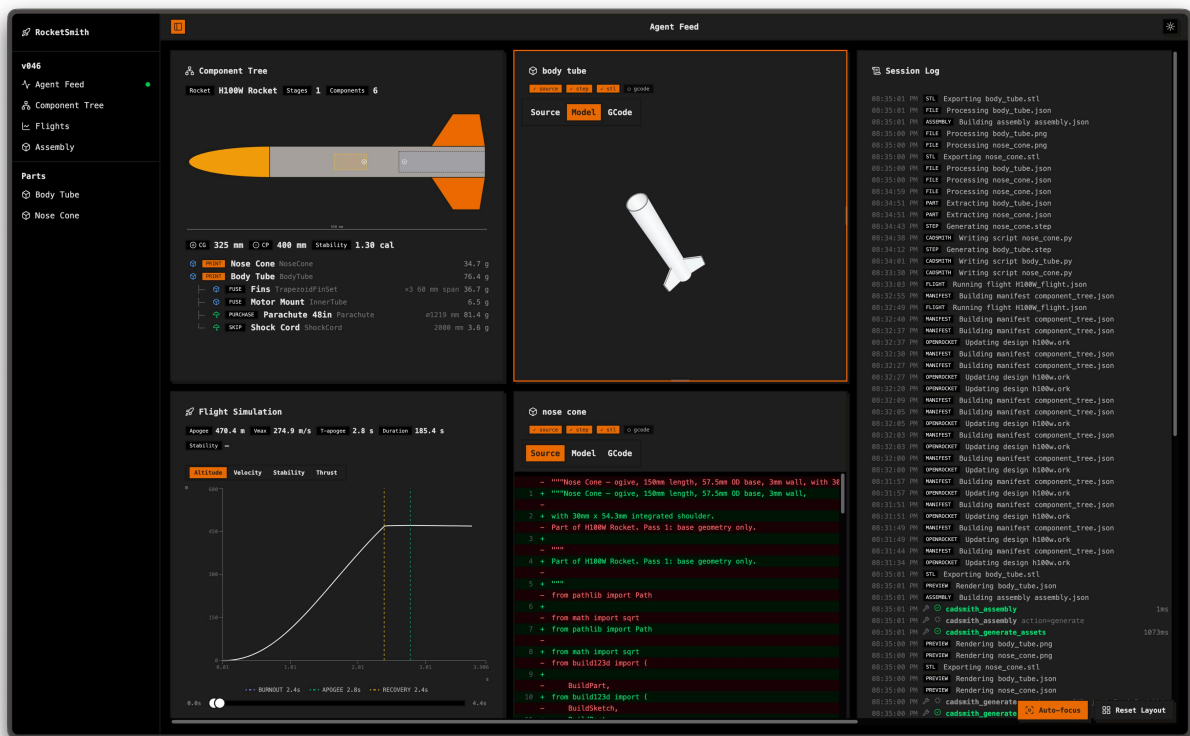


Figure 6: (Top Window) Terminal with Claude Code provides the primary means to control the RocketSmith agentic system with initial and follow up prompts for designing high powered rockets. (Bottom Window) Web based GUI displays tool calls results, flight simulations, component trees, and CADSmith model outputs. Locally deployed server to connects two windows with realtime updates allowing for visualization of key aspects of the rocket development process.

and highlighting active cards.

4.2.3 Skills

Skills provides a concrete set of instructions to provide the agent when performing a specific task and in RocketSmith a total of 7 different skills are instantiated: `design-for-additive-manufacturing`, `generate-structures`, `mass-calibration`, `modify-structures`, `motor-selection`, `print-preparation`, and `stability-analysis`. Within workflows of the RocketSmith agentic system, `motor-selection` is the first skill that is executed as this provides a base to build the designs and structures upon. Afterwards OpenRocket⁵ is performed to generate a stable design which is then verified with the `stability-analysis` skill. An optional `design-for-additive-manufacturing` skill concerned with component consolidation is performed at this stage, specifically investigating motor mounts and couplers whose parts can be combined into a single 3D printable assembly. The skill for `generate-structures` is performed to outline the CAD model for the first pass (i.e. fins, walls, etc.) and a subsequent skill of `modify-structures` is executed to add minor adjustments (i.e. tap holes, screw holes, etc.). The last `print-preparation` outlines the procedure to send the STEP files for each component over to PrusaSlicer⁷ for weight estimation and tool path generation.

4.3 Manufacturing

A total of four high powered rockets were manufactured from designs generated using the RocketSmith agentic system; referred to as High Power 1, 2, 3, and 4 (Figure 8). All airframe components were fabricated with the Fused Deposition Modeling (FDM) additive manufacturing process in either Polyethylene Terephthalate Glycol (PETG)⁵⁰ or Acrylonitrile Butadiene Styrene (ABS)⁵¹ filament. A variety of FDM printers (Figure 7) were utilized to print the various components including a Creality Ender 3, Creality Ender 5 Plus, and the custom built `Voron-2-Tall`⁵² (Appendix B). The materials of PETG and ABS were selected as opposed to Polylactic Acid (PLA)⁵³ for its relatively higher heat deflection temperatures.

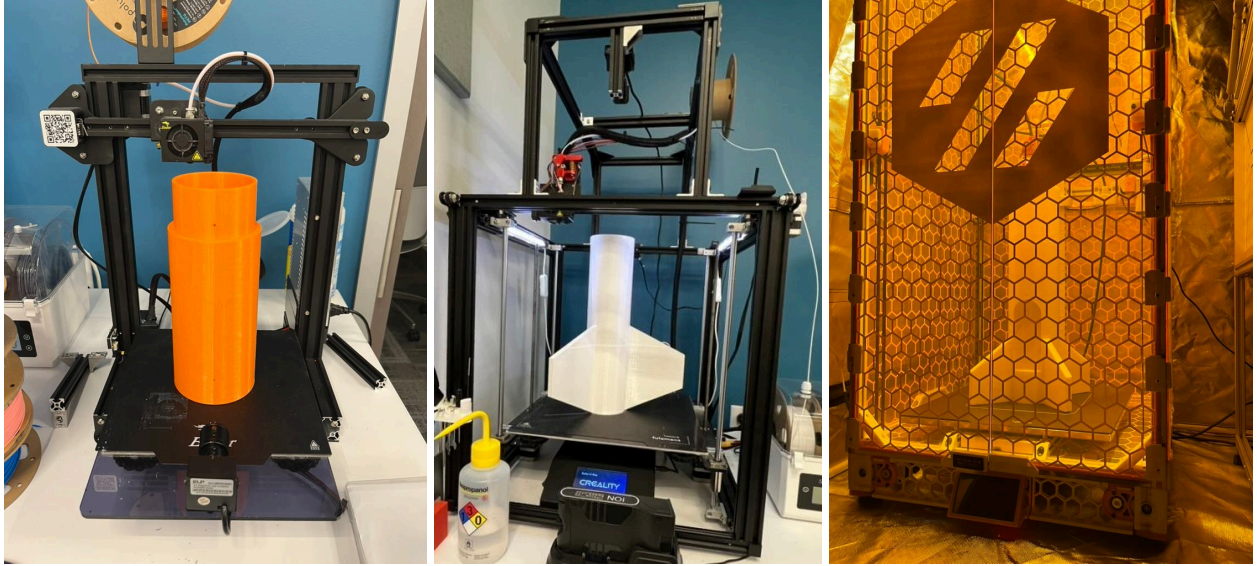


Figure 7: (Left) Ender 3 with printed middle airframe component for High Power 2. (Middle) Ender 5 Plus with printed lower airframe component for High Power 2. (Right) Voron-2-Tall⁵² with printed lower airframe component for High Power 1.

All components were printed with a 15% gyroid infill, 4 to 5 vertical walls, and a 0.20 mm layer height. Three of the four high power rockets were composed of two manufactured airframe components, those being the nose cone and airframe body. These airframe bodies extended past the printable z height of the Ender 5 Plus (400 mm) and were instead printed using the Voron-2-Tall⁵² with ABS filament. The dimensions of these nose cones were sufficiently small enough to manufacture using the Ender 5 Plus and printed with PETG filament. High Power 2 is an outlier as it was designed to be fabricated with commercial 3D printers with components dimensioned to fit within the build volumes of an Ender 5 Plus and Ender 3 printers. A combination of these two printers were used to print the nose cone, middle airframe, and lower airframe using PETG filament. During the manufacturing process consideration was given to the mating of the various components and necessary adjustments were made to ensure adequate fit between the various components.

4.4 Assembly

Figure 8 displays four high powered rockets that were designed and assembled by their respective engineers: Pak, Loghmani, and Barkley. Pak designed and constructed two separate high power rockets: High Power 1 (Section 4.4.1) with a level 1 impulse classification and High Power 4 (Section 4.4.4) with a level 2 impulse classification. Loghmani designed and built High Power 2 (Section 4.4.2), a level 1 impulse high powered rocket designed for manufacturing using the commercially available Creality Ender 5 Plus and Creality Ender 3 printers. Barkley also designed and assembled a level 1 impulse high powered rocket, High Power 3 (Section 4.4.3), with simplified designs enabled through large format FDM printing. Further characteristics and quantities for each rocket are outlined in Table 1.

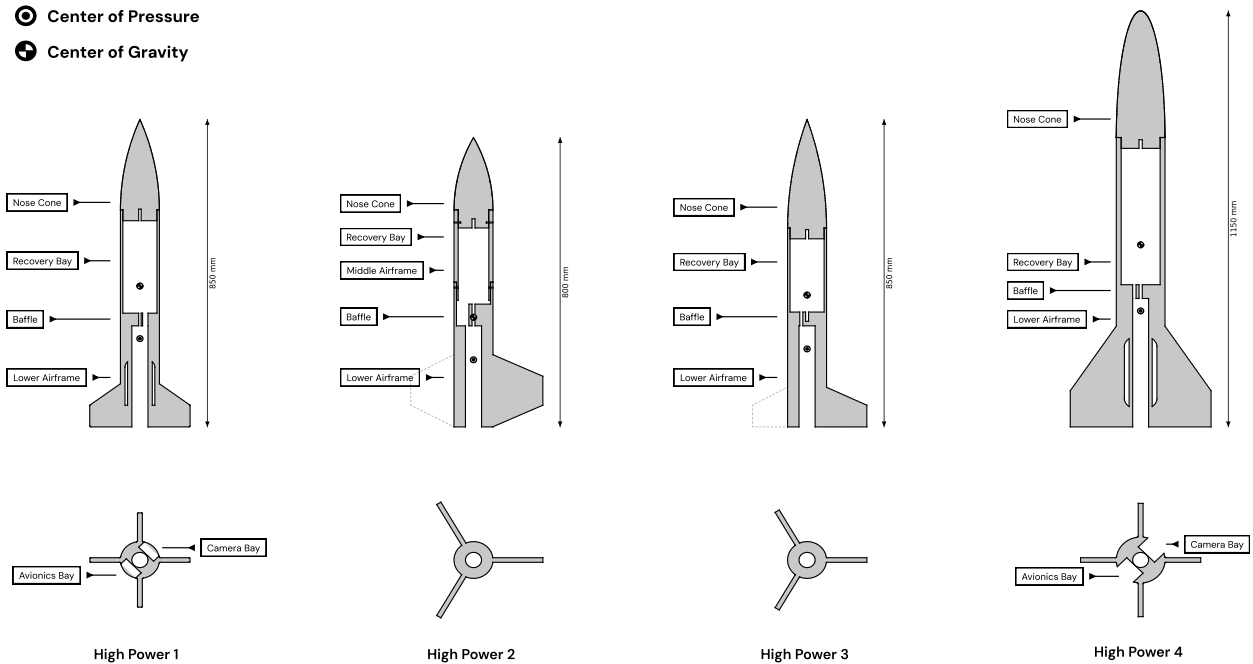


Figure 8: Rocket design schematics for High Power 1 (Pak), 2 (Loghmani), 3 (Barkley), and 4 (Pak) along with their respective components and bays. The recovery bay for High Power 2 is constructed using a separate middle airframe component in order to fit onto Creality Ender 5 Plus print bed. High Power 1 and 4 include slots to house altimeters and cameras for data recording during flight testing.

Table 1: Specific characteristics and quantities for High Power 1, 2, 3, and 4

Quantity	High Power 1	High Power 2	High Power 3	High Power 4
Developer	Pak	Loghmani	Barkley	Pak
Motor	H100W	H219T	H219T	J425R
Recovery	36" Parachute	48" Parachute	48" Parachute	48" Parachute
Altimeter	StratoLogger CF	N/A	N/A	StratoLogger CF
Camera	RunCam 5	N/A	N/A	RunCam 5
Length	850 mm	800 mm	850 mm	1150 mm
Diameter	101.6 mm	101.6 mm	101.6 mm	127.0 mm
Fins	4	3	3	4
Printed Parts	4	3	2	6
Materials	ABS & PETG	PETG	ABS & PETG	ABS & PETG
k Shear Pins	0	4	0	0
Mass	2249.8 g	2166.09 g	1857 g	5158.8 g
Mass (dry)	1988.8 g	1905.09 g	1596 g	4527.8 g
Stability	1.26 cal	1.63 cal	1.27 cal	1.52 cal
Stability (dry)	1.46 cal	1.82 cal	1.47 cal	1.68 cal

4.4.1 High Power 1

High Power 1 (Figure 9) designed and constructed by Pak, is a single motor deploy level 1 high powered rocket composed of two primary components of a lower airframe and nose cone. This rocket, *v38* short for version 38 produced with RocketSmith, was designed around the use of a 38 mm AeroTech H100W solid rocket motor and recovery with a 36" parachute. A pair of 5/16" steel eyebolts were attached to the base of the nose cone and the center of the baffle in the lower airframe, providing anchor points for the kevlar shock cord used to connect the recovery parachute to the rest of the body. The lower airframe was printed with Voron-2-Tall FDM printer⁵² using ABS filament with slots cut out for StratoLogger CF altimeter and a RunCam 5 camera for flight data recording. The nose cone was printed with the Creality Ender 5 Plus using PETG filament.

4.4.2 High Power 2

High Power 2, *Ruminator*, (Figure 10) is a single-stage high power rocket with a three component assembly designed and constructed by Loghmani. It used an AeroTech H219T solid

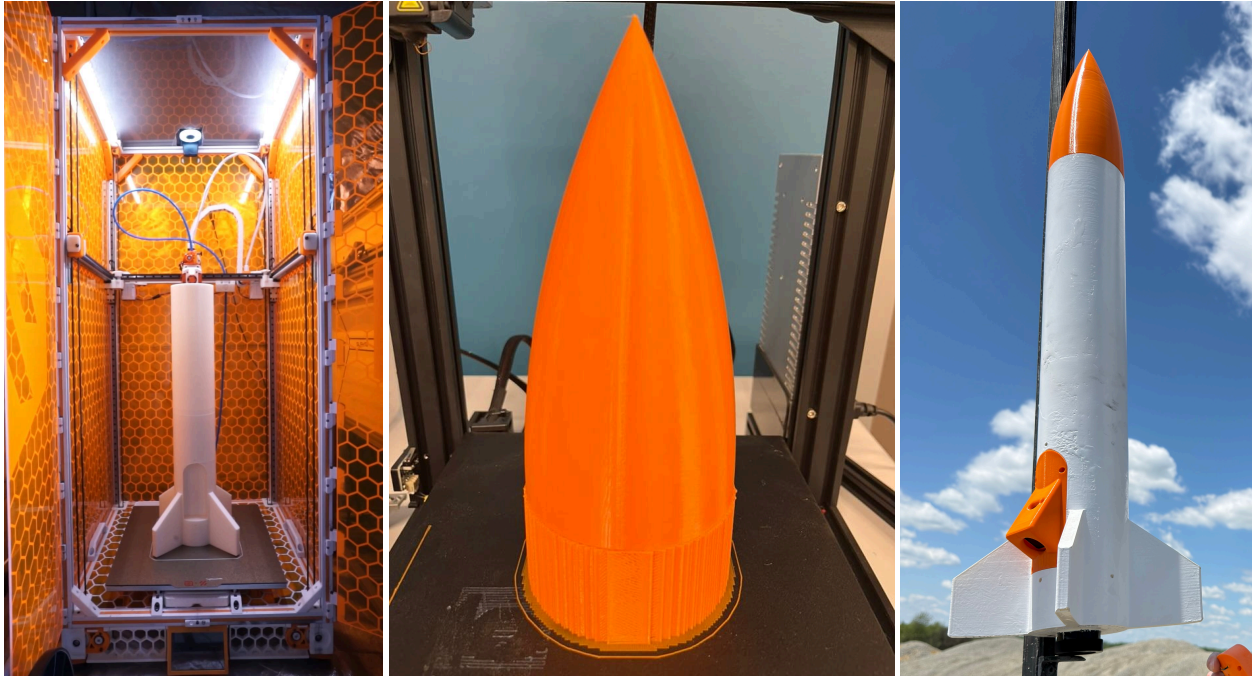


Figure 9: (Left) Finished lower airframe printed using ABS filament over the course of several days (Middle) Nose cone printed using PETG with supports to account for the overhang above the shoulder (Right) Assembled *v38* rocket prepared for launch on 1010 rail.

rocket motor and a 48" parachute. During development, RocketSmith was given the motor specification and a three-body airframe requirement, and from this generated a trapezoidal three-fin design and inner diameter constraints. The three parts consist of the lower airframe, middle airframe, and nose cone. The rocket was 31.5 inches in length and the body's outer diameter was 4 inches with a 0.425 inch wall thickness. The ogive nose cone and middle airframe were each 7.87 inches in length, and the lower airframe was 15.75 inches in length. The nose cone and middle airframe were designed to be printed on a Creality Ender 3, and the lower airframe on a Creality Ender 5 Plus. The lower airframe had a three-legged baffle design to hold the eyebolt that was attached to the parachute, and the nose cone similarly had a hole in the bottom to hold an eyebolt. A shock cord was attached to each eyebolt with a 48" parachute attached and packed inside the middle airframe for deployment at apogee. The three-part design simplified eyebolt installation. The baffle's gap allowed the apogee ignition charge to separate the nose cone and middle airframe parts from the lower airframe.

To ensure the nose cone and middle airframe stayed attached, heat inserts and screws were added. The lower airframe and middle airframe were connected via four shear pins, designed to shear when the apogee ignition charge activates. RocketSmith was used to adjust the CAD model and create pilot holes for the heat inserts, shear pins, and eyebolts. To join all parts, RocketSmith also added 1.97 inch shoulders between the lower and middle airframes, and between the middle airframe and the nose cone.

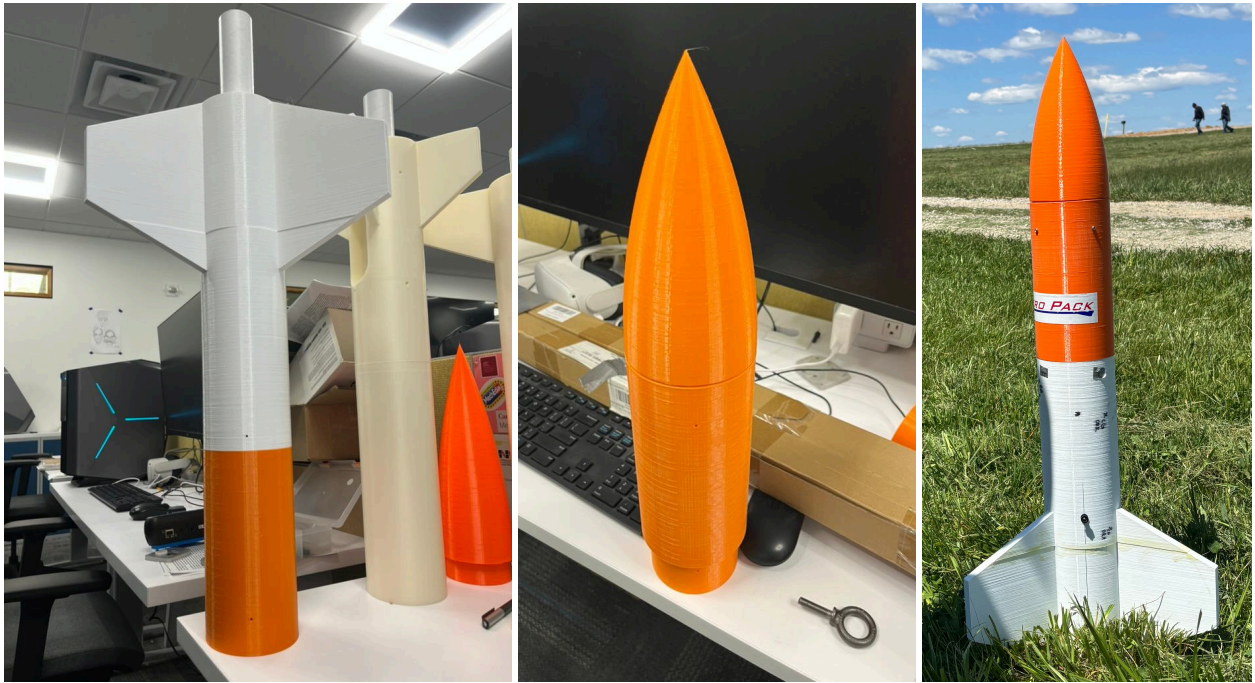


Figure 10: (Left) 38 mm motor tube inserted into the lower airframe to ensure adequate fit (Middle) Assembled nose cone and middle airframe components secured with M3 bolts and heat inserts (Right) Completely assembled rocket at launch site.

4.4.3 High Power 3

High Power 3 (Figure 11), the *H219T Thunderbolt*, was the simplest of the case studies, comprised of only a nose cone and a single main airframe rather than the segmented nose-cone, middle-airframe, and lower-airframe arrangement used in High Power 2. The body tube was around 4 inches (101.6 mm) in outer diameter with a quarter-inch (6.35 mm) wall printed in ABS, chosen for its higher heat resistance during motor ignition relative to PLA

or PETG. An AeroTech H219T solid rocket motor seated in a printed motor-mount region whose wall was locally thickened and integrated into the main airframe rather than authored as a separate inner tube. The nose cone was retained by a friction-fit shoulder rather than by shear pins as this method applies less resistance during motor ejection. For recovery anchoring, the CADSmith⁶ agent placed a pilot hole at the center of the underside of the nose cone shoulder and a second pilot hole at the center of the integrated ejection-charge baffle inside the main airframe. A 5/16" steel eyebolt was attached to both ends of the rocket providing a connection to the lower airframe and nose cone via shock cord. A 48 inch parachute was attached and packed forward of the baffle for recovery deployment.



Figure 11: (Left) Lower airframe printed using ABS filament with motor tube inserted (Right) Initial completed assembly of rocket with post design installation of rail buttons.

4.4.4 High Power 4

High Power 4 (Figure 12), *Big Heavy 2*, is a level 2 impulse class rocket designed around an AeroTech J425R single use solid rocket motor. Of the four rockets, it is the largest with a lower airframe height of 800 mm and an overall height of 1150 mm. Similar to High Power

1, it has slots cut out to house a StratoLogger CF altimeter and a RunCam 5 camera within the lower airframe. The rocket utilizes a 48" parachute for the recovery phase and utilizes two steel eyebolts to connect the nose cone and lower airframe via kevlar shock cord. The covers for the electronics along with the motor retainers are designed and printed outside the scope of the RocketSmith agentic system and printed with PETG on the Creality Ender 3 printer.

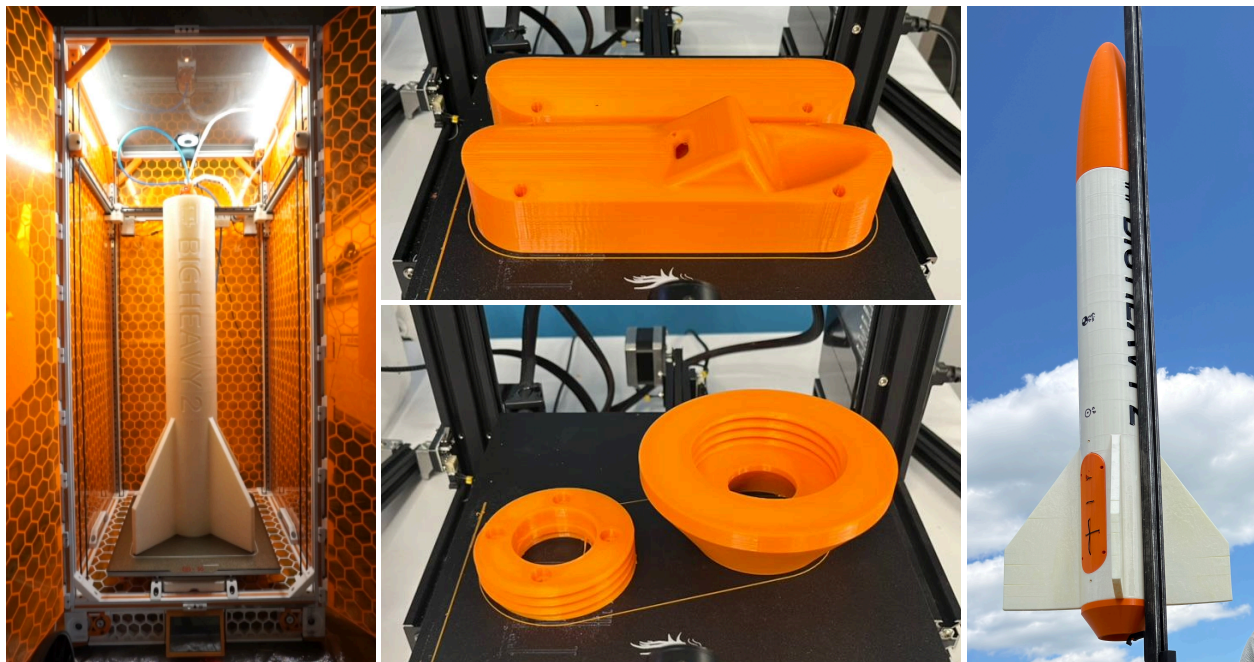


Figure 12: (Left) Finished lower airframe component printed with Voron-2-Tall⁵² using ABS filament over a duration of 4 days (Top Middle) Covers for altimeter and camera components along with (Bottom Middle) motor retainer components printed with PETG filament using Creality Ender 3 printer (Right) Assembled *Big Heavy 2* prepared on launch rail.

5 Results

5.1 Flight Tests

Flight tests of all High Power rockets were performed on Sunday, May 3rd 2026 at *Dragon's Fire Field*, an hour drive south of Pittsburgh in Fayette County. Clear sky and moderate

winds (around 10 mph) from the west enabled nearly ideal flight conditions during the permitted operating window of 9 AM to 5 PM (Appendix D). All flight tests were performed under the supervision of members from the Pittsburgh prefecture of the Tripoli Rocketry Association. Preflight checks of all rockets were done by the Range Safety Officer (RSO) and all suggested adjustments were applied accordingly before flight testing.

Table 2: Expected maximum altitude, velocity, and acceleration values along with measured data points for rockets with altimeters (High Power 1 and High Power 4).

		High Power			
		1	2	3	4
Apogee	Expected	338.8 m	334 m	473.9 m	570.1 m
	Measured	276 m	N/A	N/A	479 m
Max Velocity	Expected	76.4 m/s	94.1 m/s	114.5 m/s	115.2 m/s
	Measured	63.3 m/s	N/A	N/A	92.4 m/s
Max Acceleration	Expected	48.4 m/s ²	128.4 m/s ²	147.7 m/s ²	78.8 m/s ²
	Measured	26.9 m/s ²	N/A	N/A	54.8 m/s ²

High Power 1 - 3 were simultaneously launched within the same volley with High Power 4 launched by itself on a later volley (Figure 2). Of the four rockets, two were successfully recovered in re-flyable condition (High Power 3 and High Power 4) whereas High Power 1 encountered structural issues from the ejection charge and High Power 2 encountered recovery deployment issues. Collected data from High Power 1 and High Power 4 showed that both reached apogees close to that of the predicted flight simulation (80% and 84% of expected altitude respectively).

5.1.1 High Power 1

High Power 1 was launched successfully in the first volley alongside High Power 2 and High Power 3. Its various components were recovered damaged and in non-re-flyable condition due to a catastrophic structural failure within the lower airframe. The onboard electronics including the StratoLogger CF altimeter (Figure 17) and the RunCam 5 camera (Figure 19) were successful in recording flight data and recovered in reusable condition. Data collected from the altimeter shows that the rocket reached a measured maximum altitude of 276 m



Figure 13: (Top Left) Recovered High Power 1 components after fracture from ejection charge (Bottom Left) Recovered High Power 2 components after recovery bay separation failure (Middle) Successfully recovered High Power 3 after landing in tree (Right) High Power 4 after successful recovery deployment and landing in field.

(905 ft), around 80% of its expected apogee of 338 m (1108 ft). Recovered components of High Power 1 can be seen in Figure 13 where a clear split is visible in sections of the lower airframe midway through the baffle and just above the motor tube (Figure 14). At recovery it was visible that the eyebolt of the lower airframe was dislodged from its installed location within the baffle causing the bottom half lower airframe to return to the ground in free fall conditions. The top portion of the lower airframe and nose cone landed gently with the successful recovery deployment of the 36" parachute.

5.1.2 High Power 2

High Power 2 was tested alongside High Power 1 and High Power 3 without onboard instrumentation. The launch was vertical and visibly stable, without any weathercocking or tipping during the boost phase. The recovery ejection charge is believed to have detonated at apogee but separation of the recovery bay did not occur and the rocket returned to the ground at high velocity. High Power 2 hit the ground with ballistic impact and embedded itself approximately 4 meters into the soil, embedding components of the nose cone and



Figure 14: Recovered components of High Power 1 after successful launch including: (Left to Right) damaged nose cone with eyebolt removed post recovery, top section of lower airframe component with split at baffle section, middle section of lower airframe split between baffle and fin can, and bottom section of lower airframe including housing for electronics.

middle airframe. The force of landing detached the lower airframe from the middle airframe and nose cone, and the baffle detached from the lower airframe, lodging itself deep into the middle airframe and covering the parachute and shock cord. Post-flight inspection confirmed that the lower airframe's fins remained intact, although cracks along the body of the lower airframe exposed the internal print infill.

5.1.3 High Power 3

High Power 3 was the first successful launch and recovery of a high powered rocket designed by RocketSmith. At launch the rocket displayed visual stability with no off-axis tipping during the boost phase. Although no flight data was recorded, visually estimated apogee was consistent with what was expected from the flight simulation. The rocket entered recovery stage shortly after reaching apogee and experienced significant descent time with the deployed 48" parachute. Due to the lengthy descent period, the rocket drifted and was lodged in a tree, however landed intact and in reflyable condition (Figure 13). In order to recover the rocket, the trees surrounding and holding the rocket were cut and during this retrieval the lower airframe of the rocket split. To the best of the authors knowledge, the failures within the lower airframe occurred during the retrieval phase and did not occur during the

flight of High Power 3.



Figure 15: Recovered High Power 3 rocket after removal from tree with lower airframe split occurring during the retrieval.

5.1.4 High Power 4

High Power 4 was the second successful launch and recovery of a level 2 impulse classification rocket developed using RocketSmith. It was launched in its own separate volley and recovered successfully in reflyable condition with minimal effort. The onboard StratoLogger CF altimeter measured an apogee of 479 m (1571 ft), 84% of the expected 570 m (1870 ft). Separation of the recovery bay occurred approximately at apogee and deployment of the 48" parachute was performed successfully. Due to the single deployment of recovery components the main parachute also carried the components a significant distance where it was easily recovered in a farm field. Video of the entire flight and descent was captured using the onboard RunCam 5 camera (Figure 20).

6 Discussion

6.1 Design and Manufacturing

RocketSmith enables the automated design and manufacturing of high powered rockets, however during the development process of the flight tested high power rockets, various steps required input from the authors. Within the design space, High Power 1 and High



Figure 16: Recovered High Power 4 rocket in reflyable condition with minor cosmetic blemishes along the nose cone and lower airframe.

Power 4 utilized manually designed CAD components. The lower airframe and nose cone components of High Power 1 and High Power 4 were designed solely with the RocketSmith agentic system with human guidance. Using RocketSmith, slots for electronics were cut into the lower airframe in anticipation for covers designed at a later stage. These covers were designed outside of the RocketSmith agentic system due to relative complexity of the desired design. In addition, the motor retainer for High Power 4 was designed separately using SolidWorks and later integrated into the rest of the rocket.

For all rockets, the integrated PrusaSlicer slicer platform was used primarily for weight estimation of each component. Actual slicing and printing of each component's STEP file was executed outside the RocketSmith agentic system with print profiles suitable for the each design and printer (i.e. inclusion of supports and print speed). This decision to perform manual slicing reduced the risk of print failures allowing the authors to adhere to the tight manufacturing schedule before launch. Specific to components printed with ABS, cracking was observed through various parts of the airframe, especially in High Power 1 and High Power 4. These structural defects were fixed post build using epoxy to fill in the gaps and provide more structural integrity. Approaches to address this issue could utilize in-situ monitoring techniques^{16,54-56} to detect these issues but a more immediate solution would be to increase the insulation of the build chamber and potentially utilize a less temperature sensitive material such as PETG. For future versions of RocketSmith, the utilization

of PrusaSlicer will be developed to allow for greater visual validation and customization, enabling a truly end-to-end pipeline.

The most time consuming aspect of high power rocket development is the consistent iteration to designs and simulations to ensure suitable values regarding stability and recovery deployment. A zero-shot approach is feasible with RocketSmith, however there are many additional considerations during development that promote a human in the loop utilization of this agentic system. Regardless of which approach is taken, this system has shown to significantly reduce the time and friction of the successful development and testing of high powered rockets.

6.2 Flight Tests

Of flight tests, electronics were installed on two of the four high powered rockets. These are the rockets that were developed by Pak who had previously earned a Tripoli Rocket Association Level 1 Certification and best allocated the limited electronics to maximize the chance of recovery. As such, the performed flight tests were primarily concerned with evaluating the design and manufacturing capability of RocketSmith where each rocket was shown to be successful in achieving stable launch. With instrumentation installed into High Power 1 and High Power 4, further insight into each of their respective flights can be extracted.

High Power 1 reached around 80% of its expected apogee, however, the recovery phase deviates significantly from the expected trajectory (Figure 17). From the altimeter and the onboard video recording, the motor ejection charge detonation was observed around 14 seconds after launch. This is significantly past the expected motor ejection of 10 seconds just past the point in time where the rocket is expected to reach apogee. With the delayed separation of the recovery bay, the rocket experienced significantly higher forces from the deployment of the 36" parachute attributing to the catastrophic failure of the lower airframe. Root Cause Corrective Action (RCCA) for this issue would involve more precise delay charge drill adjustment to reduce potential deviation at the time of recovery bay separation. Another

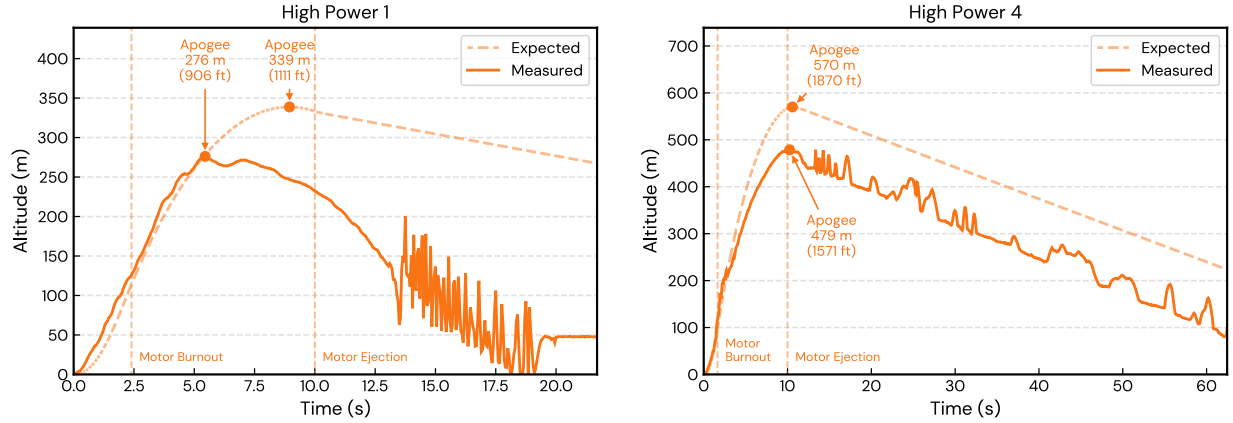


Figure 17: Flight data collected from StratoLogger altimeters on High Power 1 and High Power 4 plotted alongside expected altitudes. (Left) High Power 1 reaches around 80% of expected altitude split of the lower airframe resulted in a noisy altitude reading towards the end of flight. (Right) High Power 4 reaches approximately 84% of expected apogee and altimeter readings indicate a smooth descent resulting in a successful recovery of rocket in reflyable condition.

option would be to use an altimeter based approach prescribed to deploy recovery devices at a period just after apogee and further enabling reliability through redundancy.

High Power 4 performed near to the expected flight trajectory with the recovery bay separation occurring at almost exactly the measured apogee. The measured apogee is approximately 84% of the expected with the descent rate matching that of the expected from flight simulation (Figure 17). This enabled an easy successful recovery of the rocket with all components remaining intact and in reflyable condition. Compared to all the tested high power rockets, High Power 4 performed the best however did have the most human involvement during the development.

Performed flight tests show that the manufactured high powered rocket reflect properties closely to the designs produced by the RocketSmith agentic system. Flight characteristics such as stability, weight, and apogee are metrics that are critical to a successful launch, however other considerations regarding manufacturing and assembly are necessary for a successful recovery. Factors regarding manufacturing and other specialized high power rocketry knowledge would further enable the agentic system to construct designs capable of reliable

launch and reuse.

7 Conclusion

Flight tests performed using High Power 1, 2, 3, and 4 show that RocketSmith is capable of the automated development of additively manufactured high powered rockets. With the use of subagents and skills, the agentic system is able to outline suitable high power rocket blueprints, design for additive manufacturing, generate CAD models, and optimize flight critical values through subsequent iterations. The outputs from RocketSmith are additively manufactured with a variety of FDM printers and flight tested at a launch event where two of the four tested high power rockets were successfully recovered in reflyable condition. All rockets achieved a stable launch and onboard instrumentation showed an apogee accuracy of 80% and 84% for High Power 1 and High Power 4 respectively. These results show that an agentic system is capable of successfully designing complex assemblies suitable for additive manufacturing, validated on domain specific applications such as high powered rocketry.

Appendix A Motor Impulse Classification

Motor classification is determined by the total impulse delivered by the propulsion system; specifically, this work utilizes a series of solid composite propellant. Total impulse is defined by the thrust integrated over the burn time.⁵⁷ Equation 2 defines the thrust force F in Newtons as the product of mass flow rate of propellant $\dot{m}$ in kg/s and the exhaust velocity v_e in m/s .^{11,57} The integral of the thrust force for the burn duration t_b established the total impulse I_{total} of the motor.^{11,57}

$$F = \dot{m}v_e, \quad I_{total} = \int_0^{t_b} F(t) dt \quad (2)$$

Specific to amateur rocket activities, the Federal Aviation Administration (FAA) officially designates 3 classes of rockets: Class 1 - Model Rocket, Class 2 - High Power Rockets, and Class 3 - Advanced High Power Rockets.³ Tripoli Rocketry Association and National Association of Rocketry provide further rocket motor codes for total impulse ranges (A - O) where the letter specifies the total impulse, intermediate number specifies the average thrust in newtons and the last number specifies the time delay between motor burnout and recovery ejection (i.e. a motor with designation H100W-14 defines a motor with a total within the range of 160.01 - 320 Newton-seconds, an average thrust of 100 newtons, a manufacturer specific white color, and a maximum time delay of 14 seconds). Table 3 in Appendix A outlines the various motor classifications by total impulse range which double with each subsequent letter code and organization specific certification levels (1 - 3) are required to purchase each respective high powered motor. Most amateur rocketry activities are designated as Class 2 High Power Rockets which limits motors to a combined total impulse of 40,960 Newton-seconds (H to O motors).^{3,10}

Tripoli Rocketry Association and National Association of Rocketry require proof of certification for the purchasing and launching of high powered rockets. Certifications are granted for successful launch and recovery of a rocket under the organization's specified conditions.¹⁰

Table 3: Rocket motor classification by total impulse (Classes A–O).

Class	Total Impulse (N·s)	Certification Required
A	1.26 – 2.50	None
B	2.51 – 5.00	None
C	5.01 – 10.0	None
D	10.01 – 20.0	None
E	20.01 – 40.0	None
F	40.01 – 80.0	None
G	80.01 – 160	None
H	160.01 – 320	Level 1
I	320.01 – 640	Level 1
J	640.01 – 1,280	Level 2
K	1,280.01 – 2,560	Level 2
L	2,560.01 – 5,120	Level 2
M	5,120.01 – 10,240	Level 3
N	10,240.01 – 20,480	Level 3
O	20,480.01 – 40,960	Level 3

Appendix B Voron-2-Tall

Voron-2-Tall⁵² is a modified Voron 2.4⁵⁸ FDM printer with an extended z axis build height maximum of 930 mm, particularly useful for printing tall parts such as rocket airframe components. This machine was built from a kit with original parts source for a Voron 2.4⁵⁸ with dimensions of 350 mm x 350 mm x 330 mm. Custom 2020 aluminum extrusions (1130 mm), MGN9H linear rails (1000 mm), GT2 belts, and enclosure panels (483 mm x 1103 mm x 3 mm) were sourced to extend the z axis over a range of 1000 mm. With the application of frame braces, the build height limit is reduced from 930 mm to 850 mm. Additional modifications were made including the incorporation of a *DragonBurner* toolhead for a smaller form factor along with the usage of CANBUS for the reduction of wires and removal of cable chains.



Figure 18: (Left) **Voron-2-Tall** during assembly, upside down to access internal electronics underneath the build plate. (Middle) First successful *Benchy* print with fully assembled printer and *StealthBurner*⁵⁹ toolhead using PLA filament. (Right) **Voron-2-Tall** with finished print for lower airframe component of *Big Heavy 2*⁶⁰ using ABS filament.

Appendix C Onboard Camera Data



Figure 19: View of the launch site taken from High Power 1 onboard RunCam 5 camera just before reaching an apogee of 338 m (1108 ft).



Figure 20: Photo taken using RunCam 5 from High Power 4 around measured apogee of 570 m (1870 ft).

Appendix D Launch Site Weather Conditions

Table 4: Hourly weather data during Sunday May 3rd, 2026 launch window at Fayette County, PA.

Time	Temperature	Cloud	Humidity	Wind		
				Speed	Direction	Gust
9 AM	41°F	18%	72%	5.8 mph	WSW	11.2 mph
10 AM	45°F	19%	60%	7.8 mph	W	17.0 mph
11 AM	48°F	76%	49%	9.6 mph	W	18.7 mph
12 PM	51°F	29%	43%	10.3 mph	W	19.6 mph
1 PM	53°F	18%	39%	10.5 mph	W	20.3 mph
2 PM	54°F	14%	37%	9.6 mph	W	20.6 mph
3 PM	55°F	13%	37%	9.6 mph	WNW	19.9 mph
4 PM	56°F	15%	38%	9.6 mph	W	20.2 mph
5 PM	56°F	50%	40%	8.9 mph	W	19.9 mph

References

- (1) Yang, S.; Zhao, Y. F. Additive manufacturing-enabled design theory and methodology: a critical review. *The International Journal of Advanced Manufacturing Technology* **2015**, *80*, 327–342.
- (2) Blakey-Milner, B.; Gradl, P.; Snedden, G.; Brooks, M.; Pitot, J.; Lopez, E.; Leary, M.; Berto, F.; du Plessis, A. Metal additive manufacturing in aerospace: A review. *Materials & Design* **2021**, *209*, 110008.
- (3) Federal Aviation Administration Requirements for Amateur Rocket Activities. 2008; <https://www.govinfo.gov/link/fr/73/73781>.
- (4) anthropics/claude-code. 2026; <https://github.com/anthropics/claude-code>, original-date: 2025-02-22T17:41:21Z.
- (5) Niskanen, S. OpenRocket. 2015; <https://github.com/openrocket/openrocket>, original-date: 2012-09-29T16:50:37Z.
- (6) Barkley, J.; Loghmani, R.; Farimani, A. B. CADSmith: Multi-Agent CAD Generation with Programmatic Geometric Validation. 2026; <http://arxiv.org/abs/2603.26512>, arXiv:2603.26512 [cs].
- (7) prusa3d/PrusaSlicer. 2026; <https://github.com/prusa3d/PrusaSlicer>, original-date: 2016-03-01T14:26:57Z.
- (8) Choi, Y. C.; Noh, K.-H.; Lee, J.-W.; Byun, Y.-H.; Park, B.-K. Optimal air-launching rocket design using system trades and a multi-disciplinary optimization approach. *Aerospace Science and Technology* **2009**, *13*, 406–414.
- (9) Chiesa, S.; Di Sciuva, M.; Testore, L. Launch vehicles conceptual design and structural analysis: an integrated approach via FEM. *Aircraft Design* **1999**, *2*, 117–145.

- (10) Stine, G. H.; Stine, B. *Handbook of model rocketry*, 7th ed.; Wiley: Hoboken, NJ, 2004.
- (11) Sutton, G. P.; Biblarz, O.; Morehart, J. H. *Rocket Propulsion Elements*, 10th ed.; Wiley, 2026.
- (12) Ock, J.; Sharma Meda, R.; Vinchurkar, T.; Jadhav, Y.; Farimani, A. B. Adsorb-Agent: autonomous identification of stable adsorption configurations via a large language model agent. *Digital Discovery* **2026**, *5*, 617–629.
- (13) Ock, J.; Guntuboina, C.; Barati Farimani, A. Catalyst Energy Prediction with CatBERTa: Unveiling Feature Exploration Strategies through Large Language Models. *ACS Catalysis* **2023**, *13*, 16032–16044.
- (14) Bartsch, A.; Farimani, A. B. LLM-Craft: Robotic Crafting of Elasto-Plastic Objects With Large Language Models. *IEEE Robotics and Automation Letters* **2025**, *10*, 10450–10457.
- (15) Merrill, C.; Raman, A.; George, A.; Barati Farimani, A. LLM-drone: aerial additive manufacturing with drones planned using large language models. *Construction Robotics* **2025**, *9*, 20.
- (16) Jadhav, Y.; Pak, P.; Barati Farimani, A. LLM-3D print: Large Language Models to monitor and control 3D printing. *Additive Manufacturing* **2025**, *114*, 105027.
- (17) Pak, P.; Barati Farimani, A. AdditiveLLM: Large language models predict defects in metals additive manufacturing. *Additive Manufacturing Letters* **2025**, *14*, 100292.
- (18) Pak, P.; Chandrasekhar, A.; Barati Farimani, A. Agentic additive manufacturing alloy evaluation. *Additive Manufacturing Letters* **2026**, *17*, 100355.
- (19) Han, K.; Maddikayala, S.; Knappe, T.; Patel, O.; Liao, A.; Barati Farimani, A. TDFlow: Agentic Workflows for Test Driven Development. Proceedings of the 19th Conference

- of the European Chapter of the Association for Computational Linguistics (Volume 1: Long Papers). Rabat, Morocco, 2026; pp 1511–1527.
- (20) He, J.; Treude, C.; Lo, D. LLM-Based Multi-Agent Systems for Software Engineering: Literature Review, Vision, and the Road Ahead. *ACM Trans. Softw. Eng. Methodol.* **2025**, *34*, 124:1–124:30.
 - (21) Nigam, V.; Chandrasekhar, A.; Barati Farimani, A. Polymer-Agent: Large Language Model Agent for Polymer Design. *Journal of Chemical Information and Modeling* **2026**, *66*, 5145–5154.
 - (22) Chaudhari, A.; Ock, J.; Barati Farimani, A. Modular large language model agents for multi-task computational materials science. *Communications Materials* **2026**,
 - (23) Jadhav, Y.; Barati Farimani, A. Large language model agent as a mechanical designer. *Journal of Engineering Design* **2026**, *0*, 1–37, _eprint: <https://doi.org/10.1080/09544828.2026.2624356>.
 - (24) Yao, S.; Zhao, J.; Yu, D.; Du, N.; Shafran, I.; Narasimhan, K.; Cao, Y. REACT: SYNERGIZING REASONING AND ACTING IN LANGUAGE MODELS. 2023.
 - (25) Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, L.; Polosukhin, I. Attention is All you Need. *Advances in Neural Information Processing Systems*. 2017.
 - (26) Simonds, T. LLMs for Engineering: Teaching Models to Design High Powered Rockets. 2025; <http://arxiv.org/abs/2504.19394>, arXiv:2504.19394 [cs] version: 1.
 - (27) Ceotto, G. H.; Schimitt, R. N.; Alves, G. F.; Pezente, L. A.; Carmo, B. S. RocketPy: Six Degree-of-Freedom Rocket Trajectory Simulator. 2021; <https://github.com/RocketPy-Team/RocketPy>, Issue: 6 Publication Title: Journal of Aerospace Engineering Volume: 34 original-date: 2018-09-10T00:21:57Z.

- (28) Sifan Wu; Amir Khasahmadi; Mor Katz; Pradeep Kumar Jayaraman; Yewen Pu; Karl Willis; Bang Liu CAD-LLM: Large Language Model for CAD Generation. 2023.
- (29) Li, X.; Sun, Y.; Sha, Z. LLM4CAD: Multimodal Large Language Models for Three-Dimensional Computer-Aided Design Generation. *Journal of Computing and Information Science in Engineering* **2024**, *25*.
- (30) contributors, C. CadQuery. 2026; <https://zenodo.org/doi/10.5281/zenodo.3955118>.
- (31) Open-Cascade-SAS/OCCT. 2026; <https://github.com/Open-Cascade-SAS/OCCT>, original-date: 2020-03-16T15:04:06Z.
- (32) Barrowman, J. The Practical Calculation of the Aerodynamic Characteristics of Slender Finned Vehicles. 2013.
- (33) Bhat, B. N. *Aerospace Materials and Applications*, 1st ed.; Progress in Astronautics and Aeronautics Ser; American Institute of Aeronautics & Astronautics: Reston, 2018.
- (34) Nakano, R. et al. WebGPT: Browser-assisted question-answering with human feedback. *ArXiv* **2021**,
- (35) Gage, P. A new algorithm for data compression. *The C Users Journal archive* **1994**,
- (36) Liu, H.; Li, C.; Wu, Q.; Lee, Y. J. Visual Instruction Tuning. *Advances in Neural Information Processing Systems* **2023**, *36*, 34892–34916.
- (37) Shi, H.; Xu, Z.; Wang, H.; Qin, W.; Wang, W.; Wang, Y.; Wang, Z.; Ebrahimi, S.; Wang, H. Continual Learning of Large Language Models: A Comprehensive Survey. *ACM Computing Surveys* **2025**, *58*, 120:1–120:42.
- (38) Liu, N. F.; Lin, K.; Hewitt, J.; Paranjape, A.; Bevilacqua, M.; Petroni, F.; Liang, P. Lost in the Middle: How Language Models Use Long Contexts. *Transactions of the Association for Computational Linguistics* **2024**, *12*, 157–173.

- (39) Schick, T.; Dwivedi-Yu, J.; Dessi, R.; Raileanu, R.; Lomeli, M.; Hambro, E.; Zettlemoyer, L.; Cancedda, N.; Scialom, T. Toolformer: Language Models Can Teach Themselves to Use Tools. *Advances in Neural Information Processing Systems* **2023**, *36*, 68539–68551.
- (40) Yang, J.; Jimenez, C. E.; Wettig, A.; Lieret, K.; Yao, S.; Narasimhan, K.; Press, O. SWE-agent: agent-computer interfaces enable automated software engineering. Proceedings of the 38th International Conference on Neural Information Processing Systems. Red Hook, NY, USA, 2024; pp 50528–50652.
- (41) google-gemini/gemini-cli. 2026; <https://github.com/google-gemini/gemini-cli>, original-date: 2025-04-17T17:04:31Z.
- (42) openai/codex. 2026; <https://github.com/openai/codex>, original-date: 2025-04-13T05:37:54Z.
- (43) Wu, Q.; Bansal, G.; Zhang, J.; Wu, Y.; Li, B.; Zhu, E.; Jiang, L.; Zhang, X.; Zhang, S.; Liu, J.; Awadallah, A. H.; White, R. W.; Burger, D.; Wang, C. AutoGen: Enabling Next-Gen LLM Applications via Multi-Agent Conversations. 2024.
- (44) Roger Maitland build123d: A Python-based parametric CAD library. 2025; <https://github.com/gumyr/build123d>.
- (45) openrocket/orhelper. 2026; <https://github.com/openrocket/orhelper>, original-date: 2024-05-13T11:49:18Z.
- (46) slic3r/Slic3r. 2026; <https://github.com/slic3r/Slic3r>, original-date: 2011-09-01T19:05:05Z.
- (47) Wei, J.; Bosma, M.; Zhao, V.; Guu, K.; Yu, A. W.; Lester, B.; Du, N.; Dai, A. M.; Le, Q. V. Finetuned Language Models are Zero-Shot Learners. 2021.

- (48) Kojima, T.; Gu, S. S.; Reid, M.; Matsuo, Y.; Iwasawa, Y. Large language models are zero-shot reasoners. *Proceedings of the 36th International Conference on Neural Information Processing Systems*. Red Hook, NY, USA, 2022; pp 22199–22213.
- (49) Mosqueira-Rey, E.; Hernandez-Pereira, E.; Alonso-Rios, D.; Bobes-Bascaran, J.; Fernandez-Leal, A. Human-in-the-loop machine learning: a state of the art. *Artificial Intelligence Review* **2023**, *56*, 3005–3054.
- (50) Szykiedans, K.; Credo, W.; Osiński, D. Selected Mechanical Properties of PETG 3-D Prints. *Procedia Engineering* **2017**, *177*, 455–461.
- (51) Tanikella, N. G.; Wittbrodt, B.; Pearce, J. M. Tensile strength of commercial polymer materials for fused filament fabrication 3D printing. *Additive Manufacturing* **2017**, *15*, 40–47.
- (52) Pak, P. ppak10/Voron-2-Tall. 2026; <https://github.com/ppak10/Voron-2-Tall>, original-date: 2023-11-27T20:25:20Z.
- (53) Tymrak, B. M.; Kreiger, M.; Pearce, J. M. Mechanical properties of components fabricated with open-source 3-D printers under realistic environmental conditions. *Materials & Design* **2014**, *58*, 242–246.
- (54) Pak, P.; Ogoke, F.; Polonsky, A.; Garland, A.; Bolintineanu, D. S.; Moser, D. R.; Arnhart, M.; Madison, J.; Ivanoff, T.; Mitchell, J.; Jared, B.; Salzbrenner, B.; Heiden, M. J.; Barati Farimani, A. ThermoPore: Predicting part porosity based on thermal images using deep learning. *Additive Manufacturing* **2024**, *95*, 104503.
- (55) Ogoke, F.; Pak, P.; Myers, A.; Quirarte, G.; Beuth, J.; Malen, J.; Barati Farimani, A. Deep learning for melt pool depth contour prediction from surface thermal images via vision transformers. *Additive Manufacturing Letters* **2024**, *11*, 100243.

- (56) Bostan, B.; Hinnebusch, S.; Anderson, D.; To, A. C. Accurate detection of local porosity in laser powder bed fusion through deep learning of physics-based in-situ infrared camera signatures. *Additive Manufacturing* **2025**, *101*, 104701.
- (57) Terry W. McCreary PhD. *Experimental Composite Propellant*, 2nd ed.; GRAPHIX & STUFF, 2021.
- (58) VoronDesign/Voron-2. 2026; <https://github.com/VoronDesign/Voron-2>, original-date: 2018-11-12T23:57:59Z.
- (59) VoronDesign/Voron-Stealthburner. 2026; <https://github.com/VoronDesign/Voron-Stealthburner>, original-date: 2022-07-23T16:40:29Z.
- (60) Pak, P. ppak10/Big-Heavy. 2026; <https://github.com/ppak10/Big-Heavy>, original-date: 2023-10-15T22:11:07Z.