Tom Weering 5009057

Report 60d
Preliminary design of a parallelepiped-shaped tank for CubeSat green mono-propellant systems

Problem Formulation

As space miniaturization advances, it becomes increasingly important to integrate diverse fields of knowledge, such as green fuels, advanced manufacturing techniques, and innovative design shapes, to drive progress in the sector. This project focuses on the analysis and preliminary design of a CubeSat propellant tank with a parallelepiped shape, utilizing additive manufacturing (AM) techniques for green monopropellants. The goal is to design a competitive design solution, taking note of their respective advantages and disadvantages (Table 1).

Designing a parallelepiped-shaped propellant tank for CubeSats introduces both challenges and opportunities, especially when incorporating AM techniques and green monopropellants. The parallelepiped geometry offers space efficiency and modularity, which simplifies spacecraft integration. However by it's nature it is not an ideal shape for a pressure vessel which will limit the internal pressure.

AM also presents multiple benefits for CubeSat propulsion systems, including rapid prototyping and simplyfing the manufacturing process. It also enables the creation of more complex geometries. AM also supports high-performance alloys, such as Ti-6Al-4V and Inconel [14], ensuring structural integrity and chemical compatibility with green monopropellants. Nevertheless, challenges remain, such as post-processing requirements for flight-grade standards, print orientation, thermal issues, the need for supports etc.

Green monopropellants offer improved safety, lower toxicity, reduced environmental impact, and competitive performance compared to traditional propellants. Additionally, they exhibit better storage stability. However, due to their relatively recent development and limited testing, it is important to consider their compatibility with the selected structural material, propulsion management systems, and other related components.

Advantages	**Disadvantages**
Parallelepiped-Shaped Tanks	
Space Efficiency Modularity	Sharp corners Pressure limited
Additive Manufacturing	
Rapid Prototyping Complex Geometries	Surface Finish Chemical Compatibility
Green (Mono-)Propellant	
Safety Storage Stability	Chemical Compatability Limited Knowledge

Table 1: Key Advantages and Disadvantages of Parallelepiped-Shaped Tanks, Additive Manufacturing, and Green Monopropellants.

The success of this project hinges on a comprehensive understanding of the interaction between these factors, with the goal of designing a propellant tank that aligns with our current knowledge of these interactions. The design will aim to provide a solution that is easily integrable into existing CubeSat formats (primarily 3U, as detailed later) and leverages the capabilities of current additive manufacturing (AM) techniques, ensuring compatibility with green monopropellants. In doing so this report aspires to answer the following two research questions:

RQ1: What is the most suitable design for a parallelepiped-shaped tank, to meet the identified constraints and requirements?

RQ2: What are the most suitable materials, dimensions and shape/fluidic interfaces which allow for a fully operational design of the tank?

Requirements

Before exploring potential design options and developing a preliminary design, it is essential to first define the problem through a set of requirements. This was achieved through an internal literature review, including an analysis of the Lunar Flashlight mission [12], reference to design manuals for pressure vessels [4], and research into constraints associated with additive manufacturing [1] and the use of green monopropellants [15]. Following a systems engineering approach, top-level requirements were established and subsequently broken down into related sub-requirements, resulting in the requirements discovery tree illustrated in Figure 1.

The requirements highlighted in light green represent stakeholder requirements, derived directly from the project description. Below these are system-level requirements, followed by more specific requirements that pertain to "subsystems" of the tank being designed. These subsystems include structural aspects, propellant management devices, CubeSat design specifications, and constraints related to additive manufacturing.

Each requirement is assigned a unique ID in the format PAMGMT-STK-XX-SYS-SUB-XX, where SUB represents one of the four subsystems: STR (Structure), PMD (Propellant Management Device), CDS (CubeSat Design Specification), or AM (Additive Manufacturing). The acronym PAMGMT stands for Parallelepiped-shaped Additively Manufactured Green Monopropellant Tank. This structured approach ensures clear traceability, linking each requirement to higher-level system goals and demonstrating how individual requirements contribute to fulfilling broader, less strictly defined objectives. The complete overview listing every requirement along with the associate rationale for including them is given in the Table below (see Table 3).

An implicit requirement is that the selected material must be compatible with L-PBF or EB-PBF additive manufacturing processes and chemically resistant to green monopropellants. According to literature [14], several materials have been identified as suitable for these manufacturing techniques, with some also exhibiting favorable chemical compatibility with green monopropellants. Table 2 provides an overview of these materials. These options will serve as the foundation for material selection in the subsequent preliminary design phase. In the table, "A" indicates excellent chemical compatibility with the specified green monopropellant. Throughout this report, L-PBF will be used as the default method for additive manufacturing, as EB-PBF generally involves higher costs due to the requirement for a vacuum environment [7]. However, if thermal issues arise with L-PBF, EB-PBF may be considered as an alternative (except for SS304L).

Table 2: Material Compatibility with Green Propellants [14]

MATERIAL	HNP225	LMP103S	AFM315E	FLP-106
Ti-6Al-4V	A	A	A	A
SS316L	A	A	A	A
SS304L	A	A	A	A
Inconel 625	A	A	A	A

*Only compatible with L-PBF (Laser Powder Bed Fusion) process.

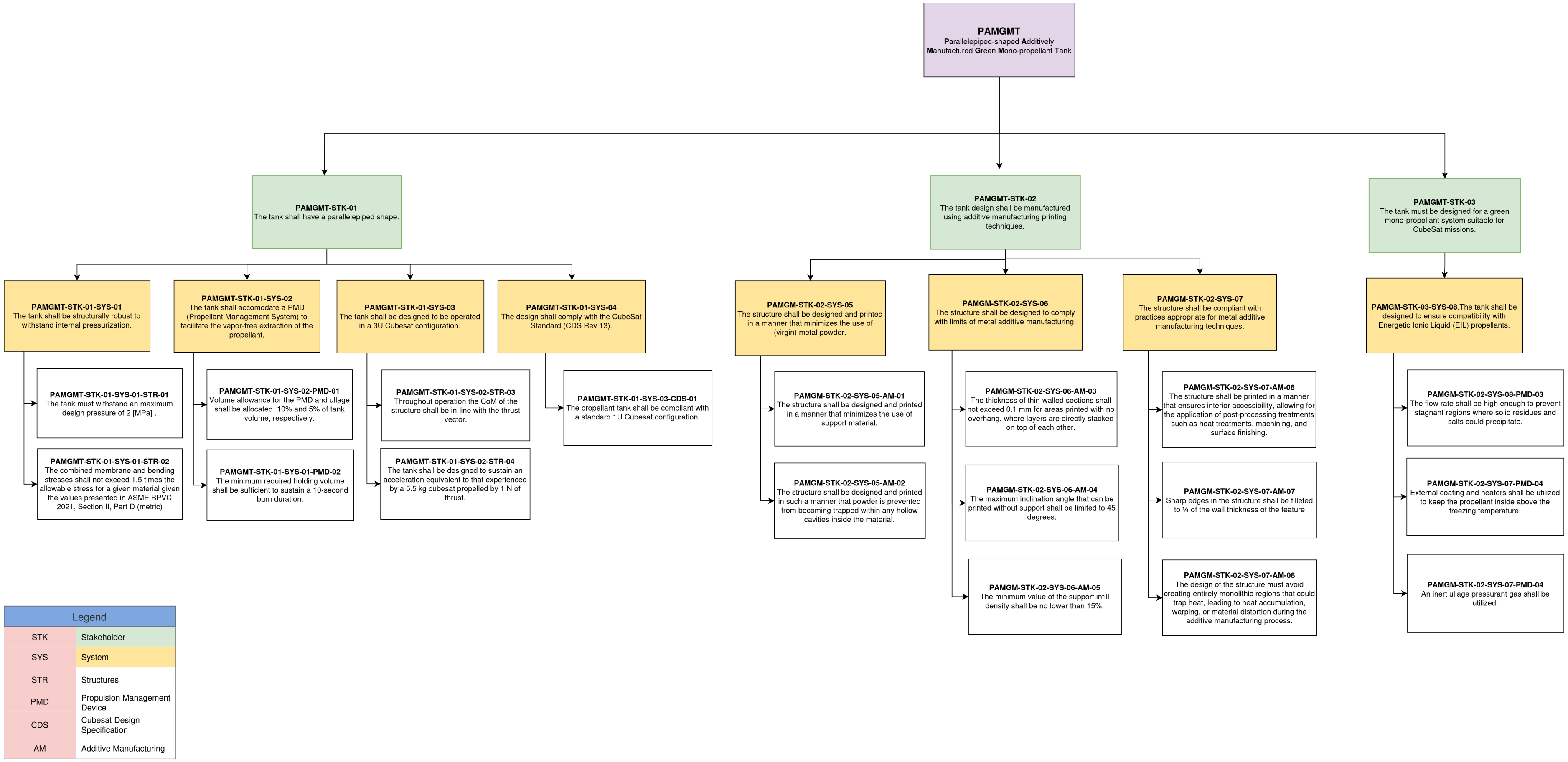

PAMGMT
Parallelepiped-shaped Additively Manufactured Green Mono-propellant Tank
PAMGMT-STK-01 The tank shall have a parallelepiped shape.
PAMGMT-STK-02 The tank design shall be manufactured using additive manufacturing printing techniques.
PAMGMT-STK-03 The tank must be designed for a green mono-propellant system suitable for CubeSat missions.
PAMGMT-STK-01-SYS-01 The tank shall be structurally robust to withstand internal pressurization.
PAMGMT-STK-01-SYS-02 The tank shall accomodate a PMD (Propellant Management System) to facilitate the vapor-free extraction of the propellant.
PAMGMT-STK-01-SYS-03 The tank shall be designed to be operated in a 3U Cubesat configuration.
PAMGMT-STK-01-SYS-04 The design shall comply with the CubeSat Standard (CDS Rev 13).
PAMGM-STK-02-SYS-05 The structure shall be designed and printed in a manner that minimizes the use of (virgin) metal powder.
PAMGM-STK-02-SYS-06 The structure shall be designed to comply with limits of metal additive manufacturing.
PAMGM-STK-02-SYS-07 The structure shall be compliant with practices appropriate for metal additive manufacturing techniques.
PAMGM-STK-03-SYS-08.The tank shall be designed to ensure compatibility with Energetic Ionic Liquid (EIL) propellants.
PAMGMT-STK-01-SYS-01-STR-01 The tank must withstand an maximum design pressure of 2 [MPa] .
PAMGMT-STK-01-SYS-01-STR-02 The combined membrane and bending stresses shall not exceed 1.5 times the allowable stress for a given material given the values presented in ASME BPVC 2021, Section II, Part D (metric)
PAMGMT-STK-01-SYS-02-PMD-01 Volume allowance for the PMD and ullage shall be allocated: 10% and 5% of tank volume, respectively.
PAMGMT-STK-01-SYS-01-PMD-02 The minimum required holding volume shall be sufficient to sustain a 10-second burn duration.
PAMGMT-STK-01-SYS-02-STR-03 Throughout operation the CoM of the structure shall be in-line with the thrust vector.
PAMGMT-STK-01-SYS-02-STR-04 The tank shall be designed to sustain an acceleration equivalent to that experienced by a 5.5 kg cubesat propelled by 1 N of thrust.
PAMGMT-STK-01-SYS-03-CDS-01 The propellant tank shall be compliant with a standard 1U Cubesat configuration.
PAMGM-STK-02-SYS-05-AM-01 The structure shall be designed and printed in a manner that minimizes the use of support material.
PAMGM-STK-02-SYS-05-AM-02 The structure shall be designed and printed in such a manner that powder is prevented from becoming trapped within any hollow cavities inside the material.
PAMGM-STK-02-SYS-06-AM-03 The thickness of thin-walled sections shall not exceed 0.1 mm for areas printed with no overhang, where layers are directly stacked on top of each other.
PAMGM-STK-02-SYS-06-AM-04 The maximum inclination angle that can be printed without support shall be limited to 45 degrees.
PAMGM-STK-02-SYS-06-AM-05 The minimum value of the support infill density shall be no lower than 15%.
PAMGM-STK-02-SYS-07-AM-06 The structure shall be printed in a manner that ensures interior accessibility, allowing for the application of post-processing treatments such as heat treatments, machining, and surface finishing.
PAMGM-STK-02-SYS-07-AM-07 Sharp edges in the structure shall be filleted to ¼ of the wall thickness of the feature
PAMGM-STK-02-SYS-07-AM-08 The design of the structure must avoid creating entirely monolithic regions that could trap heat, leading to heat accumulation, warping, or material distortion during the additive manufacturing process.
PAMGM-STK-02-SYS-08-PMD-03 The flow rate shall be high enough to prevent stagnant regions where solid residues and salts could precipitate.
PAMGM-STK-02-SYS-07-PMD-04 External coating and heaters shall be utilized to keep the propellant inside above the freezing temperature.
PAMGM-STK-02-SYS-07-PMD-04 An inert ullage pressurant gas shall be utilized.
Legend
STK Stakeholder
SYS System
STR Structures
PMD Propulsion Management Device
CDS Cubesat Design Specification
AM Additive Manufacturing

Figure 1: Requirements discovery tree for the PAMGMT.

Requirement Index	Description	Rationale
PAMGMT-STK-01	The tank shall have a parallelepiped shape.	Defined by supervisor.
PAMGMT-STK-01-SYS-01	The tank shall be structurally robust to withstand internal pressurization.	Regardless of the PMD, engine type, or propellant used, some ullage pressure is necessary. Given the tank's parallelepiped shape, special attention must be paid to ensuring structural integrity.
PAMGMT-STK-01-SYS-01-STR-01	The tank must withstand an maximum design pressure of 2 [MPa].	monopropellant microthrusters produce 0.1–1 N of thrust [6], and for a specific 1 N thruster, chamber pressure ranges from 0.45 to 2.2 MPa [15]. To accommodate various thruster types, a Maximum Design Pressure (MDP) of approximately 2 MPa was chosen.
PAMGMT-STK-01-SYS-01-STR-02	The combined membrane and bending stresses shall not exceed 1.5 times the allowable stress for a given material given the values presented in ASME BPVC 2021, Section II, Part D (metric)	The sizing must go according to standardization practices appropriate for pressure vessels. The ASME (The American Society of Mechanical Engineers) sizing for infinite rectangular pressure vessels was adapted for this purpose.
PAMGMT-STK-01-SYS-02	The tank shall accomodate a PMD (Propellant Management System) to facilitate the vapor-free extraction of the propellant.	Without settling thrust maneuvers, vapor-free propellant flow cannot be ensured without a PMD. Over time, liquid and gas phases mix, potentially covering the outlet. This can lead to combustion instabilities or even engine failure. [11]
PAMGMT-STK-01-SYS-02-PMD-01	Volume allowance for the PMD and ullage shall be allocated: 10% and 5% of tank volume, respectively.	These values are used to provide a realistic estimate of the propellant volume while accounting for the space occupied by the PMD [15].
PAMGMT-STK-01-SYS-01-PMD-02	The minimum required holding volume shall be sufficient to sustain a 10-second burn duration.	It is assumed a CubeSat burn typically lasts up to 10 seconds [6]. The PMD must supply enough propellant to sustain this duration, defining the minimum required holding volume.
PAMGMT-STK-01-SYS-03	The tank shall be designed to be operated in a 3U Cubesat configuration	Monopropellant thrusters are suitable for 3U and 12U CubeSats [6]. Since the market primarily targets 3U CubeSats, this propellant tank should be designed to align with that demand.
PAMGMT-STK-01-SYS-02-STR-03	Throughout operation the CoM of the structure shall be in-line with the thrust vector.	An asymmetrically designed PMD or an offset propellant mass will generate a moment proportional to the offset and thrust, requiring compensation from the attitude thrusters.
PAMGMT-STK-01-SYS-02-STR-04	The tank shall be designed to sustain an acceleration equivalent to that experienced by a 5.5 kg Cubesat propelled by 1 N of thrust.	A reference mass of approximately 5.5 kg was taken for a 3U CubeSat [22], which is expected to be accelerated by a 1 N thruster. This is considered suitable for a micropropulsion-driven CubeSat of this size and serves as a representative value for sizing the propellant tank.
PAMGMT-STK-01-SYS-04	The design shall comply with the CubeSat Standard (CDS Rev 13).	The CubeSat Design Specification (CDS) [21] is a widely recognized standard for CubeSat design. Complying with this standard ensures compatibility with other CubeSat components and systems.
PAMGMT-STK-01-SYS-03-CDS-01	The propellant tank shall be compliant with a standard 1U Cubesat configuration.	To create a versatile, "plug-and-play" system adaptable to various missions and configurations, the propellant tank dimensions were constrained to fit within a 1U CubeSat, ensuring adequate clearance for structural components like mounting points.
PAMGMT-STK-02	The tank design shall be manufactured using additive manufacturing printing techniques.	Defined by supervisor.
PAMGM-STK-02-SYS-05	The structure shall be designed and printed in a manner that minimizes the use of (virgin) metal powder.	Virgin metal powder will likely be required for flight parts [12], as studies suggest that reused powder can result in inconsistent material properties and performance. However, since this significantly increases costs, its use should be minimized.
PAMGM-STK-02-SYS-05-AM-01	The structure shall be designed and printed in a manner that minimizes the use of support material.	Since the supports will likely be made from the same virgin metal powder as the main structure, minimizing their use is essential to reduce material consumption.
PAMGM-STK-02-SYS-05-AM-02	The structure shall be designed and printed in such a manner that powder is prevented from becoming trapped within any hollow cavities inside the material.	Hollow structures without release holes trap powder, increasing mass and wasting material.
PAMGM-STK-02-SYS-06	The structure shall be designed to comply with limits of metal additive manufacturing.	Like any manufacturing method, additive manufacturing has unique constraints that must be considered during the design process.
PAMGM-STK-02-SYS-06-AM-03	The thickness of thin-walled sections shall not exceed 0.1 mm for areas printed with no overhang, where layers are directly stacked on top of each other.	A value of 0.1 mm is taken to be the minimum thickness that can be printed without support [23].
PAMGM-STK-02-SYS-06-AM-04	The maximum inclination angle that can be printed without support shall be limited to 45 degrees.	A value of 45 degrees is taken to be the maximum angle that can be printed without support [23].
PAMGM-STK-02-SYS-06-AM-05	The minimum value of the support infill density shall be no lower than 15%.	Support structures can be printed at a minimum distance of 2mm from eachother without causing thermal problems [23], a simple estimation on the available infill material for the supports can lead to a value as low as 5% but to remain conservative a value of 15% which is often utilized in conventional plastic 3D printing is used.
PAMGM-STK-02-SYS-07	The structure shall be compliant with practices appropriate for metal additive manufacturing techniques.	The design must align with metal additive manufacturing requirements to ensure successful fabrication, including post-processing considerations and proper handling after production.
PAMGM-STK-02-SYS-07-AM-06	The structure shall be printed in a manner that ensures interior accessibility, allowing for the application of post-processing treatments such as heat treatments, machining, and surface finishing.	Post-processing treatments are often necessary to improve the mechanical properties of the printed structure [12]. Ensuring interior accessibility is essential for applying these treatments effectively.
PAMGM-STK-02-SYS-07-AM-07	Sharp edges in the structure shall be filleted to $\frac{1}{4}$ of the wall thickness of the feature.	Sharp edges on exterior components, such as ribs, must be filleted to reduce stress concentrations and enhance safety by preventing injuries during post-production handling and assembly [13].
PAMGM-STK-02-SYS-07-AM-08	The design of the structure must avoid creating entirely monolithic regions that could trap heat, leading to heat accumulation, warping, or material distortion during the additive manufacturing process.	Monolithic regions can trap heat, leading to thermal issues during the additive manufacturing process. To prevent these issues, the design must incorporate features that promote heat dissipation and prevent heat accumulation.
PAMGMT-STK-03	The tank must be designed for a green mono-propellant system suitable for CubeSat missions.	Defined by supervisor.
PAMGM-STK-03-SYS-08.	The tank shall be designed to ensure compatibility with Energetic Ionic Liquid (EIL) propellants.	Energetic Ionic Liquid (EIL) propellants are a type of green monopropellant that offers high performance and low toxicity. The tank must be designed to accommodate the unique properties of EIL propellants to ensure safe and efficient storage and delivery.
PAMGM-STK-02-SYS-08-PMD-03	The flow rate shall be high enough to prevent stagnant regions where solid residues and salts could precipitate.	A notable issue with certain EIL propellants, particularly when used in propulsion systems, is the potential for slow decomposition. This can lead to the formation of solid residues and salt precipitation, which could create operational challenges, such as clogging of valves and feed lines. To prevent this, the propellant tank must be designed to ensure adequate flow rates (value of around 30mL/min taken from [15]) that prevent stagnation and promote propellant circulation.
PAMGM-STK-02-SYS-07-PMD-04	External coating and heaters shall be utilized to keep the propellant inside above the freezing temperature.	Energetic Ionic Liquid (EIL) propellants have a low freezing point, which can lead to solidification and clogging of the propellant lines. To prevent this, the tank must be equipped with external heating elements and coatings to maintain the propellant temperature above its freezing point.
PAMGM-STK-02-SYS-07-PMD-04	An inert ullage pressurant gas shall be utilized.	To maintain the propellant in a liquid state, an inert ullage pressurant gas must be used.

Table 3: Requirements table

Design Options

To effectively define the scope of the design problem, it is essential to systematically present all possible design options within the established constraints—namely, additive manufacturing, the use of green monopropellants, and a parallelepiped-shaped configuration. This ideation process is structured through a design option tree, which aims to comprehensively explore all conceivable alternatives at this stage within the defined design space.

As a baseline, the tank design used in the Lunar Flashlight mission [12] was considered, along with hypothetical design concepts such as those proposed by Shcheglov et al. [16] and various types of additively manufactured pressure vessels discussed in Tepponen et al. [20]. Based on this, several design possibilities were identified, including a generic parallelepiped-shaped configuration (rectangular but not necessarily conforming to the CubeSat standard and potentially incorporating angled surfaces), a rounded rectangular design, a variable thickness structure, an inner lattice design, a uniform thickness rectangular design, and an externally ribbed rectangular design.

These designs come with their own benefits and drawbacks [20]:

- **Generic Parallelepiped Design**
  - **Description:** A generic rectangular tank design (i.e. follows from $V = |\vec{a} \cdot (\vec{b} \times \vec{c})|$) that can be adapted to various enclosures but lacks predefined constraints.
  - **Advantages:**
    - Can be tailored for specific enclosures.
  - **Disadvantages:**
    - Hard to design due to fewer constraints.
    - Cannot be designed as a "plug-and-play" solution.
- **Rounded Rectangular Design**
  - **Description:** A rectangular tank with rounded edges to minimize stress concentrations and improve structural durability.
  - **Advantages:**
    - Reduces stress concentrations at corners, improving fatigue life.
    - May marginally improve weight reduction in comparison to uniform thickness.
  - **Disadvantages:**
    - May slightly reduce usable internal volume due to rounded edges.
- **Variable Thickness Rectangular Design**
  - **Description:** A rectangular tank with varying wall thickness to optimize weight-to-strength ratio.
  - **Advantages:**
    - Improves weight-to-strength ratio by reinforcing high-stress areas.

    - Reduces material usage, leading to cost savings.
    - Enhances fatigue life by minimizing stress concentration zones.
  - **Disadvantages:**
    - Complex manufacturing due to varying wall thickness.
    - Requires advanced simulation and optimization tools.
    - Potential issues with residual stresses and distortions during printing.

- **Uniform Thickness Rectangular Design**
  - **Description:** A simple rectangular tank with uniform wall thickness for ease of manufacturing and predictable performance.
  - **Advantages:**
    - Simplifies manufacturing and analysis with minimal support structures.
    - Predictable mechanical behavior under pressure loads (well understood from theory [4]).
    - Easier post-processing and quality control.
  - **Disadvantages:**
    - Not optimized for weight reduction.
    - Doesn't fully leverage the potential of additive manufacturing.
- **Inner Lattice Rectangular Design**
  - **Description:** A rectangular tank with an internal lattice structure to reduce weight while maintaining structural integrity.
  - **Advantages:**
    - Enables significant weight reduction while maintaining structural integrity.
    - Difficult to manufacture using traditional methods, making AM a perfect fit.
  - **Disadvantages:**
    - Can introduce stress concentration points at lattice junctions.
    - Post-processing and inspection are more complex.
    - Inner ribs require filleting or other design modifications to prevent propellant entrapment.
- **Ribbed Rectangular Design**
  - **Description:** A rectangular tank reinforced with external or internal ribs to improve stiffness and resistance to buckling.
  - **Advantages:**
    - Increases stiffness and resistance to buckling.
    - Provides significant weight reduction while maintaining structural integrity.
    - Ribs can serve as mounting points for structural integration.
  - **Disadvantages:**
    - Requires additional supports between ribs during manufacturing.

These design options were evaluated using a trade-off matrix, considering factors such as weight reduction, design simplicity, adaptability, volume utilization, and printability. As shown in Table 4, the ribbed and uniform rectangular designs emerged as the most favorable. While they may not be the most intricate or most compelling options for showcasing additive manufacturing techniques, they offer a balanced combination of simplicity, high printability, and well-documented stress and thickness relationships [3] [4]. Therefore given the objectives of this design and analysis study these options were chosen to be implemented in the preliminary sizing.

Design	**Weight Reduction (5)**	**Design Simplicity (4)**	**Adaptable (3)**	**Volume Utilization (2)**	**Printable (1)**	**Total Score**
Generic Parallelepiped	2 (10)	1 (4)	5 (15)	5 (10)	2 (2)	41
Rounded Rectangular	3 (15)	3 (12)	3 (9)	3 (6)	4 (4)	46
Variable Thickness	4 (20)	2 (8)	3 (9)	3 (6)	1 (1)	44
Uniform Thickness	2 (10)	5 (20)	4 (12)	4 (8)	5 (5)	55
Inner Lattice	4 (20)	2 (8)	3 (9)	2 (4)	4 (4)	45
Ribbed Rectangular	4 (20)	4 (16)	4 (12)	4 (8)	4 (4)	60

Table 4: Trade-off analysis of different tank designs. Numbers in parentheses represent weighted scores.

The next set of design considerations involves selecting the most appropriate Propellant Management Device (PMD). Given PAMGMT-STK-01-SYS-02, which aims to ensure vapor-free propellant delivery to the engine, a thorough evaluation of PMD options is essential. The choice of PMD directly affects the required internal pressure, influencing skin thickness and other structural aspects of the design. The most commonly used PMDs in CubeSats include elastic diaphragms, actuated expulsion mechanisms, surface tension components, and gas pressurization [14].
These common PMD types are described and their advantages and disadvantages are presented bellow [11]:

- **Elastic Diaphragms**
  - **Description:** Utilizes a flexible membrane that separates the pressurant gas from the liquid propellant.
  - **Advantages:**
    - Prevents gas-propellant mixing, ensuring vapor-free propellant delivery.
    - Effective for slosh control.
  - **Disadvantages:**
    - Limited lifespan due to elastomeric material degradation.
    - Additional mass due to the diaphragm.
- **Actuated Expulsion Mechanisms (Pistons, Bladders etc.)**

  - **Description:** Uses some type of actuated mechanism such as pistons, bladders or even pumps to actively expel propellant.
  - **Advantages:**
    - The ability to control the propellant mass flow rate.
    - High expulsion efficiency.
    - Allows for higher propellant storage volumes by eliminating the need for gas pressurization systems (in case of using pumps).
  - **Disadvantages:**
    - Requires power.
    - Increased system mass and complexity.

- **Surface Tension Components (Capillary PMDs)**
  - **Description:** Utilizes capillary forces to control and direct liquid propellant within the tank, ensuring vapor-free expulsion under microgravity conditions.
  - **Advantages:**
    - Passive operation with no moving parts, increasing reliability.
    - Lightweight and highly adaptable to different tank geometries.
    - Effective in managing liquid positioning and preventing vapor ingestion.
  - **Disadvantages:**
    - Performance depends on precise geometric design and wettability properties.
    - Limited capability to handle high flow rates compared to active expulsion methods.
    - Complex manufacturing and integration, especially for intricate structures.
- **Gas Pressurization**
  - **Description:** Uses inert gas (e.g., helium, nitrogen) to directly pressurize the propellant.
  - **Advantages:**
    - Simple design, widely used in propulsion systems.
  - **Disadvantages:**
    - Requires large storage tanks for pressurant gases.
    - Creates larger internal pressure.

A trade-off matrix was created based on the criteria of weight reduction, design simplicity, adaptability, ullage pressure, and power consumption. The last two criteria specifically assess the additional internal pressure required by each PMD and whether the system requires power to operate (with a score of 5 indicating no power requirement and 1 representing a high, continuous power demand). The results are presented in Table 5. Surface tension methods are clearly favored due to their simplicity, lack of power requirements, and minimal impact on the necessary ullage pressure.

PMD Type	Weight Reduction (5)	Design Simplicity (4)	Adaptable (3)	Ullage Pressure (2)	Power Consumption (1)	Total Score
Elastic Diaphragms	2 (10)	3 (12)	3 (9)	2 (4)	3 (3)	38
Actuated Expulsion	3 (15)	2 (8)	5 (15)	4 (8)	1 (1)	47
Surface Tension	5 (25)	5 (20)	2 (6)	4 (8)	5 (5)	64
Gas Pressurization	4 (20)	4 (16)	3 (9)	1 (2)	2 (2)	49

Table 5: Trade-off analysis of different PMD options. Numbers in parentheses represent weighted scores.

After selecting for the PMD category, various components within the surface tension category can still be considered to ensure vapor-free propellant delivery. Table 6 provides an overview of these components, including a brief description, their advantages, and their disadvantages.

According to literature [6], monopropellant thrusters are among the most suitable propulsion choices for 3U and 12U CubeSats, with the current market primarily catering to the 3U class. The proposed tank should be compact enough to fit within the structural frame of a 1U CubeSat while being designed primarily for integration into 3U CubeSat missions. This means it must withstand the accelerations typical for such a class of spacecraft.

As a baseline, an acceleration of approximately 0.018 [g] is assumed, based on a 5.5 [kg] spacecraft mass [22] and a total thrust of about 1 [N] (see PAMGMT-STK-01-SYS-02-STR-04). Unlike the Lunar Flashlight mission, this tank is not being developed for a specific mission but rather as a versatile, multi-functional, plug-and-play solution. For most CubeSats, propulsion is primarily required for orbit adjustments and station-keeping maneuvers, typically involving burn durations of no more than 10 seconds (see PAMGMT-STK-01-SYS-01-PMD-02).

Given these constraints, the CubeSat is expected to experience low accelerations for relatively short periods. From Table 6, sponges and vanes emerge as suitable options for guiding and controlling propellant flow, with porous elements inherently incorporated within sponge structures. While baffles could help mitigate sloshing, their necessity is reduced due to the relatively small propellant mass fraction in a 3U CubeSat and the expected low acceleration levels. Therefore, at this stage of the design, sloshing is not considered a critical concern. Figure 2 shows the completed design option tree with all previously discussed options.

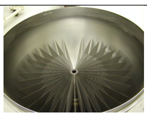
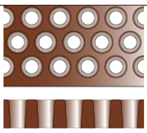
Name	Image	Description	Advantages	Disadvantages
Vane		Lightweight "sheet metal" structures that use surface tension to guide propellant to the outlet. Propellant clings to the vanes and moves along formed corners.	Simple, low cost, effective for low-acceleration burns, reliable.	Cannot lift propellant at high accelerations, limited to low-g applications.
Gallery		Closed flow paths covered with porous elements to prevent gas ingestion while allowing liquid flow. Used for vapor-free propellant delivery.	Effective at high accelerations, ensures gas-free flow, reliable for long-duration missions.	Heavy, complex, inefficient, lower reliability.
Sponge		Open metal structures with tapered gaps that control propellant flow and prevent gas intrusion. Used in stationkeeping and settling applications.	Refillable after acceleration, reliable, effective in low-g conditions.	Cannot retain liquid at high accelerations, limited liquid retention capacity.
Trap		Solid wall containers holding propellant with porous element-covered inlets, used for one-time critical burns.	Suitable for high-acceleration events, effective for gas retention.	Not passively refillable in zero-g, unsuitable for frequent maneuvers.
Trough		Uses hydrostatics to retain propellant in an acceleration field. Can be designed for refillability or omnidirectional retention.	Works in high-g conditions, useful for specific launch and reentry needs.	Heavy, expensive, not mass-efficient compared to sponges.
Porous Elements		Used in various PMDs to allow liquid flow while preventing gas penetration. Commonly made from woven screen or perforated sheet.	Ensures gas-free propellant delivery, essential for PMD performance.	Small pores increase flow resistance, difficult to manufacture and integrate.
Baffle		Structures used to reduce sloshing by limiting propellant movement within the tank.	Enhances stability, prevents unwanted fluid motion.	Does not assist in direct propellant retention, adds weight.

Table 6: Comparison of various Propellant Management Devices (PMDs). [19]

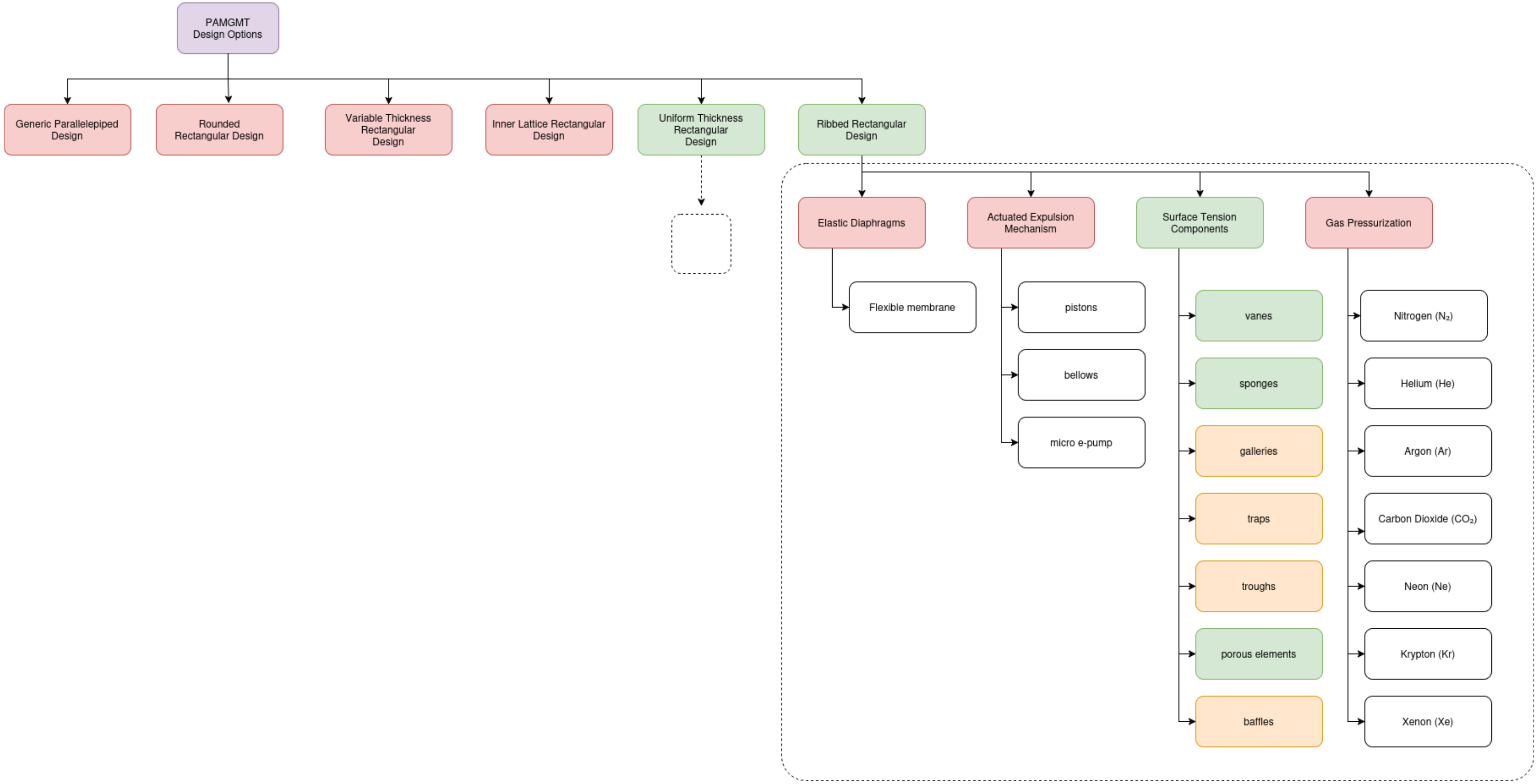


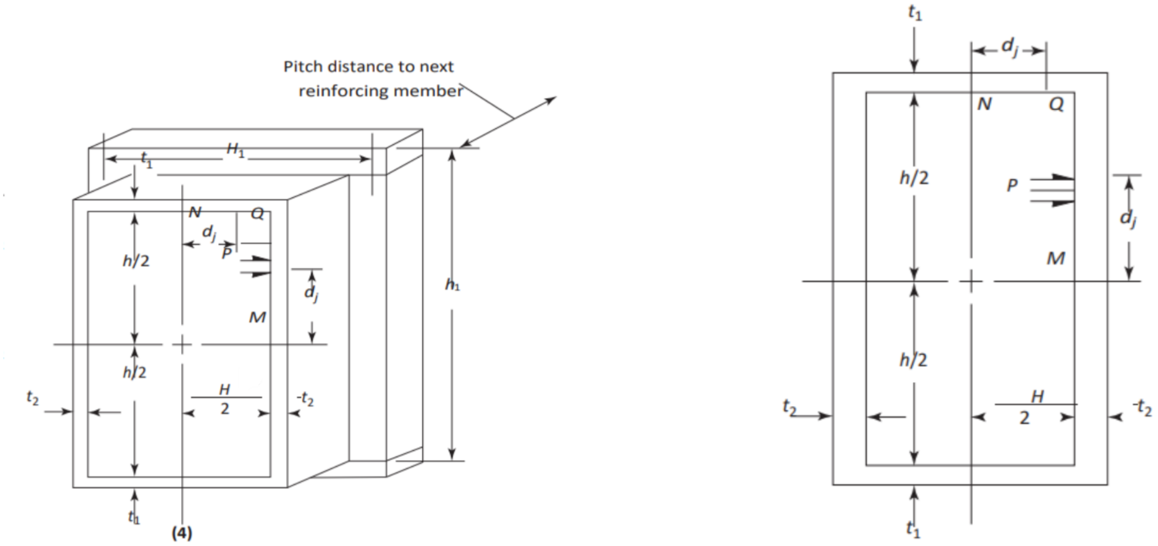
Figure 2: Design option tree

Preliminary Design

Main Structure

After specifying the design options for the tank dimensions and the PMD system, and defining the requirements, it is necessary to discuss the method used for sizing the tank. To design both a uniform thickness and ribbed rectangular pressure vessel, a method from the American Society of Mechanical Engineers (ASME) was utilized to relate the internal pressure to the required thickness and the number or spacing of ribs (i.e., stiffening elements). This method idealizes a cross-section of an infinitely long rectangular pressure vessel. Alternative methods for finite pressure vessels are described by Blach et al. [4].

The decision not to use these alternative methods (e.g., the small and large deflection methods mentioned by Blach et al.) was due to the stiffening effect induced by the corners of a 3D finite pressure vessel, which may eleviate the internal stress. To remain conservative and consistent with established standards, the ASME code method [3] was chosen. Additionally, using the ASME code offers the advantage of readily available resources and example calculations, as demonstrated by Blach et al. (calctoys). Further mass reductions could be made in the future by employing less conservative sizing methods and performing real-life pressurization validation tests. Given the chosen design options the following two generalized cross-sections were used as shown in Figure 3a and Figure 3b



(a) Cross-section of unstiffened rectangular pressure vessel.

(b) Cross-section of stiffened rectangular pressure vessel

Figure 3: Parametric rectangular presssure vessel cross-sections from ASME handbook [3]

As required by PAMGMT-STK-01-SYS-01-STR-02, the combined membrane and bending stresses shall not exceed 1.5 times the allowable stress for a given material, as specified in ASME BPVC 2021, Section II, Part D (metric) [2]. The allowable stress varies with temperature, and a simplified radiative heat balance is implemented within the code (see section) to provide a conservative estimate of the allowable stress.

The script in the code section utilizes the method outlined in Appendix 13 of the ASME handbook [3] to evaluate compliance with the specified requirement. The dimensions of the rectangular pressure vessel are predefined to fit within a 1U enclosure, as stipulated by PAMGMT-STK-01-SYS-03-CDS-01, ensuring sufficient clearance for surrounding structures (Table 8). The ribbed or stiffened configuration, illustrated in Figure 3a, is designed using predefined stiffener properties listed in Table 7.

Table 7: Summary of stiffener properties

Parameter	**Description**	**Value**
t_s	Thickness	5 mm
h_s	Height of stiffener	5 mm

Table 8: Summary of Rectangular Pressure Vessel Dimensions

Parameter	**Description**	**Value**
h	Side length	70 mm
L	Height	90 mm

The spacing between stiffeners is calculated using the following equation:

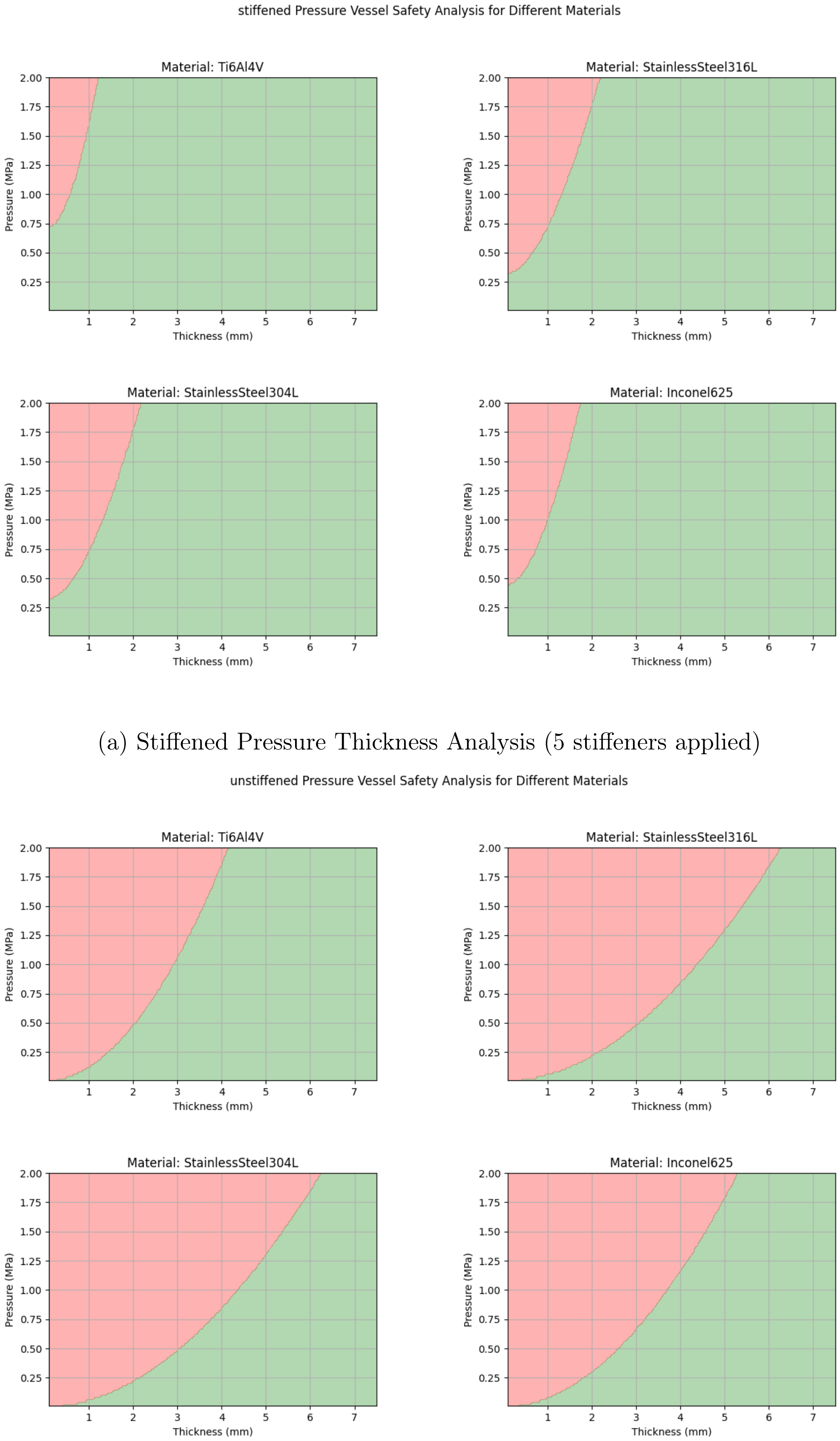
$$P = \frac{L - h_s}{n - 1} \tag{1}$$

where:

- P = Spacing between stiffeners
- L = Total height of the tank
- h_s = Height of each stiffener
- n = Number of stiffeners

Based on the material selected, as shown in Table 2, the adapted method was applied with the assumption that $t_1 = t_2$ (i.e., a uniform thickness cross-section). As illustrated in Figure 4, the safe pressure ranges are influenced by factors such as internal pressure, material properties, and the presence of stiffening elements. The results demonstrate that the stiffened structure allows for thinner walls at any given pressure, with Ti6Al4V, in particular, performing exceptionally well.

Naturally, the stiffened structure will require additional mass, and the choice of materials is also influenced by their density and the cost of the metal powder needed for construction. This relationship is illustrated in Figure 5, where the radial distance from the origin represents the viability of each design based on these two factors. The top 5 closes points to the origin are tabulated in Table 9.



(a) Stiffened Pressure Thickness Analysis (5 stiffeners applied)

(b) Unstiffened Pressure Thickness Analysis (no stiffeners applied)

Figure 4: Comparison of Pressure Thickness Analysis for Stiffened and Unstiffened Configurations

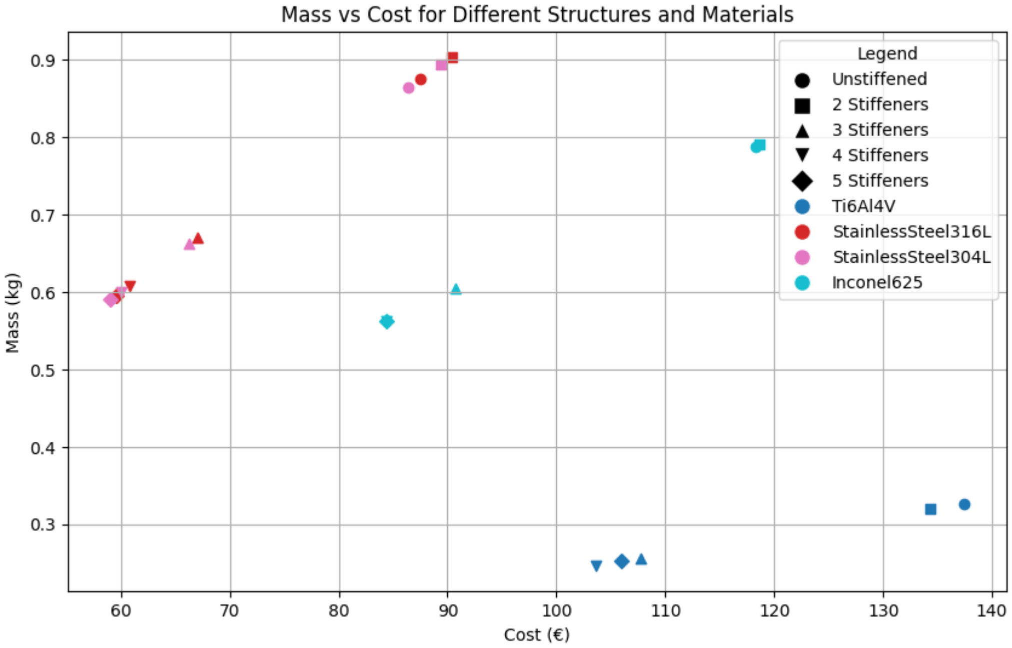


Figure 5: Mass / Cost Tradeoff for given structure type and material at a MDP of 2 [MPa]

Table 9: Top 5 Closest Points to the Origin (Mass and Cost)

Material	Structure	Mass (kg)	Cost (€)	Thickness (mm)
SS304L	5 Stiffeners	0.59	59.06	2.13
SS316L	5 Stiffeners	0.60	59.58	2.13
SS304L	4 Stiffeners	0.60	60.08	2.59
SS316L	4 Stiffeners	0.61	60.83	2.61
SS304L	3 Stiffeners	0.66	66.27	3.42

As per requirement PAMGMT-STK-01-SYS-01-STR-01, the tank is sized for a internal pressure of 2 [MPa] which will be maintained through an inert pressurant as per requirement PAMGM-STK-02-SYS-07-PMD-04. From looking at the graphs at a pressure of 2 [MPa]. It is clear that although the Titanium options are extremely lightweight, they come with the highest costs. On the other hand, the stainless steel options are the most affordable but typically have the greatest mass. The top 5 entries in the table above represent the best solutions if cost and mass were equally prioritized. However, other factors like PMD sizing, propellant requirements, and necessary supports must also be considered in the trade-off analysis. Therefore, further investigation into these aspects is necessary.

Vane sizing

As demonstrated in the previous sections, the design must include a method for extracting vapor-free propellant to meet the requirement PAMGMT-STK-01-SYS-02. Surface tension devices were selected as the most suitable solution given the constraints, with sponges and vanes being the preferred options. The vanes will be positioned at the corner

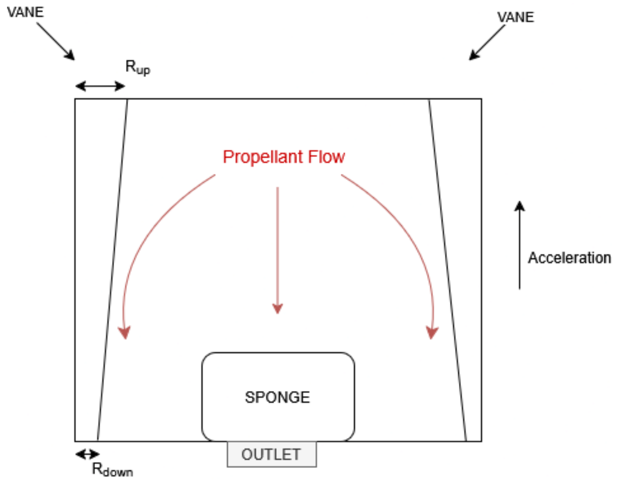


Figure 6: Diagram of PMD showing propellant flow.

points to prevent fluid from becoming trapped and to direct the propellant downward. Additionally, a central sponge will be incorporated to retain a specific amount of propellant, which will then be connected to the outlet. This has been illustrated in Figure 6.

The sizing of the vane was determined using established literature [8] [19] and is based on the following pressure balance: propellant will flow along the vane only if the "downstream radius" is sufficiently small. Essentially, the surface driving pressure is counteracted by dynamic forces, viscous losses, and hydrostatic forces. If the driving pressure is not strong enough to overcome these opposing forces, the propellant will not flow. This force balance can be represented by the following equation:

$$\sigma\left(\frac{1}{R_{down}}-\frac{1}{R_{up}}\right)=\rho(z_{down}-z_{up})+\frac{2\mu Q s^2 L}{A^3}+\frac{\rho Q^2}{2A^2} \tag{2}$$

Where:

- σ - Surface tension of the propellant
- R_{down} - Downstream radius of curvature
- R_{up} - Upstream radius of curvature
- ρ - Density of the propellant
- z_{down} - Vertical position at the downstream point
- z_{up} - Vertical position at the upstream point
- μ - Dynamic viscosity of the propellant
- Q - Volumetric flow rate of the propellant
- s - Wetting area of the vane.
- L - Length of the flow path along the vane
- A - Vane surface area.

The first term on the right-hand side represents the hydrostatic losses, the second term accounts for the viscous losses, and the third term corresponds to the dynamic pressure losses. Several assumptions and simplifications are made: first, the hydrostatic losses are neglected because the acceleration vector is largely aligned with the flow vector. Additionally, it is assumed that the wetting area is equal to the total surface area of the vane. This assumption is made because the wettability of the propellants is not fully understood and would require additional effort to model accurately. Given the limited time available, it is assumed that complete wetting will result in the maximum viscous losses.

Due to the unknown liquid properties of the propellants, such as viscosity and surface tension, similar liquids were used as references. For AF-M315E, an ethylene glycol/glycerin mixture was used; for LMP-103S, ethanol-water mixtures served as a reference; nitromethane was used for HNP225; and a nitromethane/acetonitrile mixture was used for FLP-106 [17], [5],[10]. Extra care was also taken that the overhang angle created between R_{up} and R_{down} could not surpass 45° as per requirement PAMGM-STK-02-SYS-06-AM-04.

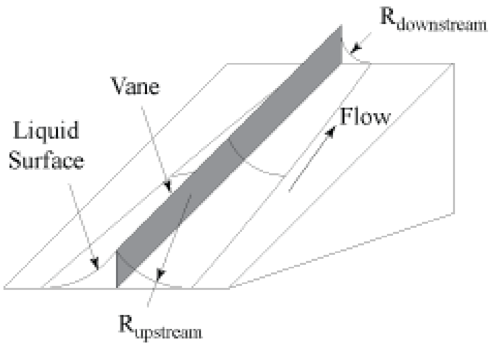


Figure 7: Vane sketch

Based on these assumptions and a constant flow rate of 30 mL/min (as per PAMGM-STK-02-SYS-08-PMD-03) regardless of the propellant type, the dynamic and viscous losses were minimal, with only a minimal difference in up-/downstream vane radius. As a result, a relatively small vane was selected using the dimensions shown in Table 10.

Table 10: Summary of Vane Radius Dimensions

Parameter	**Description**	**Value**
R_{down}	Vane radius (downstream)	0.75 mm
R_{up}	Vane radius (upstream)	0.84 mm

Additionally a spacing of around 1.5 [mm] was utilized to give enough clearance between the two vane halves and to ensure printability.

Sponge sizing

Additionally, a certain volume of propellant must be retained within a sponge. By positioning sponge panels (thin solid sheets) close together, liquid can be contained in the gaps, allowing large quantities of propellant to be managed even during thruster accelerations. At sufficiently high lateral accelerations, the propellant does not need to remain continuous and may exist in two separate regions of the sponge. This could result in isolated, inaccessible propellant within the sponge. [19]

$$a_{limit} = \frac{dg}{dz}\frac{\sigma}{\rho}\frac{2}{g^2} \tag{3}$$

- a_{limit} = limiting acceleration
- $\frac{dg}{dz}$ = gradient of gap size with respect to height
- σ = absolute surface tension
- ρ = liquid density
- g = gap size

It should be noted that the acceleration vector is assumed to align with the core of the sponge, around which the plates radiate outward (see Figure 6). This alignment suggests that the acceleration is unlikely to impose any significant constraints on the sponge's design. The sponge is sized to hold enough propellant for a 10-second burn, as specified by requirement PAMGMT-STK-01-SYS-01-PMD-02. The inner core will provide the necessary supply, while the surrounding radial plates are sized to ensure adequate taper, minimizing the likelihood of flow separation under perpendicular acceleration (should that occur during re-orientation of the spacecraft) [9].

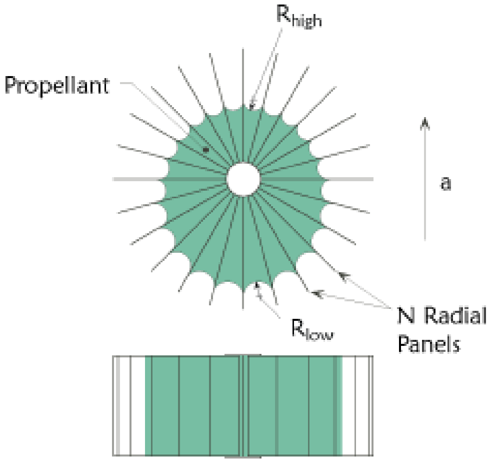


Figure 8: Sponge sketch

The thickness of the radial plates is constrained by requirement PAMGMT-STK-01-SYS-02-PMD-01. Running the main script, as shown in the code section , reveals that, based on the flow and burn time requirements, the sponges must extend up to the ceiling of the tank. It became apparent that the core would need to be quite large, and the thickness of the radial plates would be less than 1 mm. Previous work in additively manufacturing PMDs [18] highlighted the importance of aspect ratio (i.e., the height-to-thickness ratio of the radial plates) for successful printing. Without a central core to provide rigidity, the print would likely fail based on the current results. Therefore, it is crucial that the core is made of a porous material that also connects to the radial plates. This becomes even more critical since the core is too large to depend on surface tension or capillary action to prevent gas injection. A centrally located porous material helps block gas bubbles, with the bubble point pressure (ΔP) being key to the performance of the porous device. This pressure defines the minimum level required to prevent gas from passing through the barrier (see Equation (4)).

$$\Delta P_{BP} = \frac{2\sigma \cos(\alpha)}{r} \tag{4}$$

- P_{BP} = bubble point pressure
- σ = absolute surface tension
- α = contact angle
- r = pore radius

A bubble point pressure of 200 [Pa] is selected as a baseline, based on a H_2O_2 propellant system [18]. It is likely that the propellants used in this design would require a higher bubble point pressure due to their increased viscosity and surface tension compared to H_2O_2. However, increasing this pressure would necessitate a reduction in the pore radius, which would complicate the printing process of the sponge. The sponge, especially the central porous core, represents the most challenging aspect in terms of printability, making it the most difficult component to manufacture in bringing the proposed designs to reality. The feasibility of printing the porous central core with a lattice structure, based on the calculated pore radius, should be verified through a series of test prints.

Supports

In accordance with the requirement PAMGM-STK-02-SYS-05-AM-01, support material should be minimized. For the proposed structures with stiffeners, supports are necessary due to the 90-degree overhang of these elements. However, printing solid rectangular blocks from the chosen metallic powder would be inefficient. To address this, an effort was made to design a more material-efficient support structure. A 15% infill was selected in line with PAMGM-STK-02-SYS-06-AM-05. The choice of infill pattern is particularly important, as PAMGM-STK-02-SYS-05-AM-02 specifies that powder should not become trapped, nor should the pattern require excessive powder or cause any thermal problems, while still providing sufficient rigidity. Similar to sponge structures, it will be essential to validate different infill patterns through real-world test prints, with options like Gyroid infill being considered for its strength in all directions and its ability to avoid trapping powder in enclosed areas.

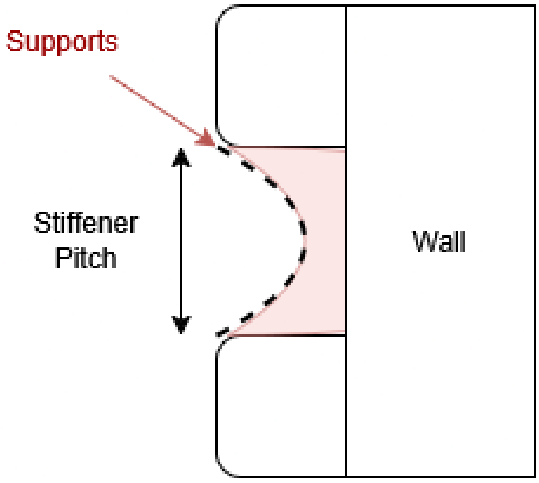


Figure 9: Sketch showing support curve in between two stiffeners

As per requirement PAMGM-STK-02-SYS-06-AM-04, overhangs of up to 45° can be printed without the need for supports. An investigation was conducted to explore the possibility of incorporating a parabolic curve within the stiffener pitch (see Figure 9). With the intention that this could reduce the amount of material needed for the supports. A simple parabolic curve was designed to fit within the available space, ensuring it extended up to the thickness of the stiffeners without exceeding a 45° angle. (Through $tan(\theta) = dy/dx$) Additionally, the "throat" of the curve was limited to 0.5 [mm] to maintain both proper printability and structural rigidity.

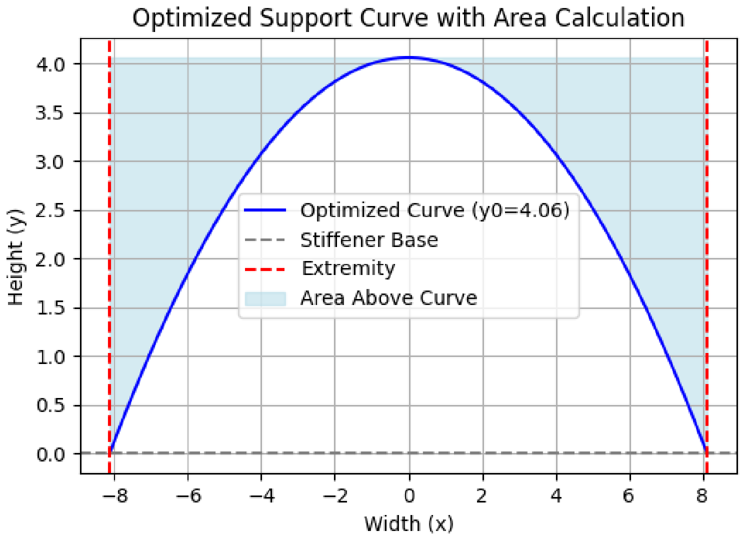


Figure 10: Optimized parabolic support curve for a 5 stiffener pitch.

In Figure 10, the optimized curve for 5 stiffener configuration is shown, the area above is calculated to be considered for the amount of support material required. Given the current proposed designs only the stiffeners necessitate these supporting structures. It is known from the literature that a theoretical overhang angle of 65° is could be feasible, suggesting that such a parabolic support structure might work. However, real-world validation tests would need to be performed to confirm this theory [23].

Trade Off

To conclude the design part of this report, the final design will be selected through a trade-off to identify the most suitable design option by evaluating different structure types, materials, and propellant types against several criteria, including the following:

- **Mass**: Lower mass is naturally preferred.
- **Effective Tank Volume**: A higher volume is desired for increased propellant capacity.
- **Cost**: Minimization of cost is preferred.
- **Printability**: This metric combines three sub-criteria:
  - Higher r_{lattice} (mm) to mitigate the issues with priting the porous central sponge material.
  - Higher t_{sponge} (mm) for better rigidity.
  - Lower support mass for reduced material usage.

The *min-max scaling* approach is used to normalize the criteria between 0 and 1. In cases where minimization is desired (e.g., mass, cost, support mass), values are inverted after normalization:

$$\text{Normalized Value} = \begin{cases} 1 - \frac{v - \min(v)}{\max(v) - \min(v)}, & \text{if minimizing} \\ \frac{v - \min(v)}{\max(v) - \min(v)}, & \text{if maximizing} \end{cases}$$

This ensures consistency in the interpretation of normalized values, where higher values are always better. The trade-off score is calculated as a weighted sum of normalized values. The chosen weights reflect the relative importance of each criterion:

- **Mass**: $w = 4$ – Highly prioritized given it's usual importance in space applications.
- **Effective Tank Volume**: $w = 1$ – Given little weight since having a rectangular tank should already allow for higher volume utilization.
- **Cost**: $w = 3$ – Given some more weight given the design should utilimately be able to compete with commercial offerings.
- **Printability**: $w = 2$ – Also given some weight considering just prioritizing weight would cause many dimensions to be very close to the printable limit.

The trade-off score is calculated as follows:

$$\text{Trade-off Score} = \frac{\sum w_i \cdot \text{Normalized Value}_i}{\sum w_i}$$

The results of this trade-off analysis are shown in Table 12, which presents the top 10 highest-rated designs out of a total of 80 variations. In contrast to the earlier mass and cost distribution, Titanium designs dominate the top 10 despite receiving low cost scores. This outcome highlights that, even with cost weighted at 3, Titanium remains the most favorable choice due to its superior performance across the other criteria.

Rank	Material	Structure	Propellant	Mass	Vol.	Cost	Print.	Score
1	Ti6Al4V	5 Stiffeners	AF-M315E	0.986	1.000	0.174	1.000	0.747
2	Ti6Al4V	4 Stiffeners	AF-M315E	1.000	0.929	0.191	0.919	0.734
3	Ti6Al4V	3 Stiffeners	AF-M315E	0.996	0.816	0.176	0.867	0.706
4	Ti6Al4V	5 Stiffeners	FLP-106	0.986	1.000	0.174	0.735	0.694
5	Ti6Al4V	5 Stiffeners	LMP-103S	0.986	1.000	0.174	0.705	0.687
6	Ti6Al4V	4 Stiffeners	FLP-106	1.000	0.929	0.191	0.648	0.680
7	Ti6Al4V	5 Stiffeners	HNP225	0.986	1.000	0.174	0.642	0.675
8	Ti6Al4V	4 Stiffeners	LMP-103S	1.000	0.929	0.191	0.621	0.674
9	SS304L	5 Stiffeners	AF-M315E	0.339	0.786	1.000	0.790	0.672
10	SS316L	5 Stiffeners	AF-M315E	0.328	0.786	0.993	0.788	0.665

Table 11: Top 10 designs with normalized values for different materials, structure types, and propellants.

The final design proposed based on the trade-off analysis is presented in Table 12. Titanium was chosen for its excellent mechanical properties, while AF-M315E was selected for its high surface tension, which enhances capillary action and enables effective liquid control in microgravity. Additionally, AF-M315E's high maturity level for space applications adds to its suitability for this design. The inclusion of stiffeners contributes to mass reduction, and the optimized support structure allows for some cost savings. However, to fully validate the proposed solutions, multiple real-world test prints are necessary.

A draft of the design is presented in Figure 11, illustrating key dimensions. It is important to note that the pores depicted in the porous core are not intended to represent the final pore structure. While the pore radius corresponds to the calculations from the sponge sizing section, various lattice structures should be explored. The representation of the porous core includes a selected number of pore circles for visualization purposes.

It is worth noting that the tank does not include top or bottom plates. Since only the cross-section was designed using the ASME code method, the end plates are assumed to have the same thickness as the walls and are intended to cover the top and bottom portion of the tank. These plates can be welded on using EB-welding, as has been done in previous similar projects [12]. This approach also ensures compliance with requirement PAMGM-STK-02-SYS-07-AM-08, which prohibits the creation of entirely monolithic structures, and requirement PAMGM-STK-02-SYS-05-AM-02, which prevents the formation of hollow cavities. Additionally, manufacturing the main walls, sponge, and top and bottom plates separately reduces potential material loss due to print failures.

Table 12: Overview of final chosen design

Property	**Value**
Material	Ti6Al4V
Structure Type	5 Stiffeners
Propellant	AF-M315E
Wall Thickness (mm)	1.18
R_{down} (mm)	0.750
R_{up} (mm)	0.840
V_{vane} (m^3)	3.41e-07
Vane Mass (kg)	1.51e-03
r_{sponge} (mm)	4.21
R_{sponge} (mm)	22.8
h_{sponge} (mm)	90.0
t_{sponge} (mm)	0.180
N_{sponge}	10
V_{sponge} (m^3)	4.01e-05
r_{lattice} (mm)	0.550
Sponge Mass (kg)	0.177
Support Mass (kg)	0.0163
Support Cost (€)	6.86
Total Mass (kg)	0.431
Total Cost (€)	188

Conclusion

In conclusion, this report aimed to answer two key research questions:

RQ1: *What is the most suitable design for a parallelepiped-shaped tank to meet the identified constraints and requirements?*
The report concluded that an externally stiffened, ribbed structure with a uniform wall thickness is the optimal solution. This design strikes a balance between a non-stiffened, uniform-thickness structure and a more complex internal lattice structure. Additionally, the ribs offer the advantage of serving as mounting points for the surrounding structure.

RQ2: *What are the most suitable materials, dimensions, and shape/fluidic interfaces to enable a fully operational tank design?*
It was determined that Titanium, specifically Ti-6Al-4V, is the most suitable material for this application. Despite its higher cost, Titanium's excellent mechanical properties and compatibility with green monopropellants and additive manufacturing make it ideal for the task. Regarding fluidic interfaces, surface tension devices provide a simple yet effective method for guiding and storing propellant. This approach has been deemed sufficient for the types of missions this tank is designed for.

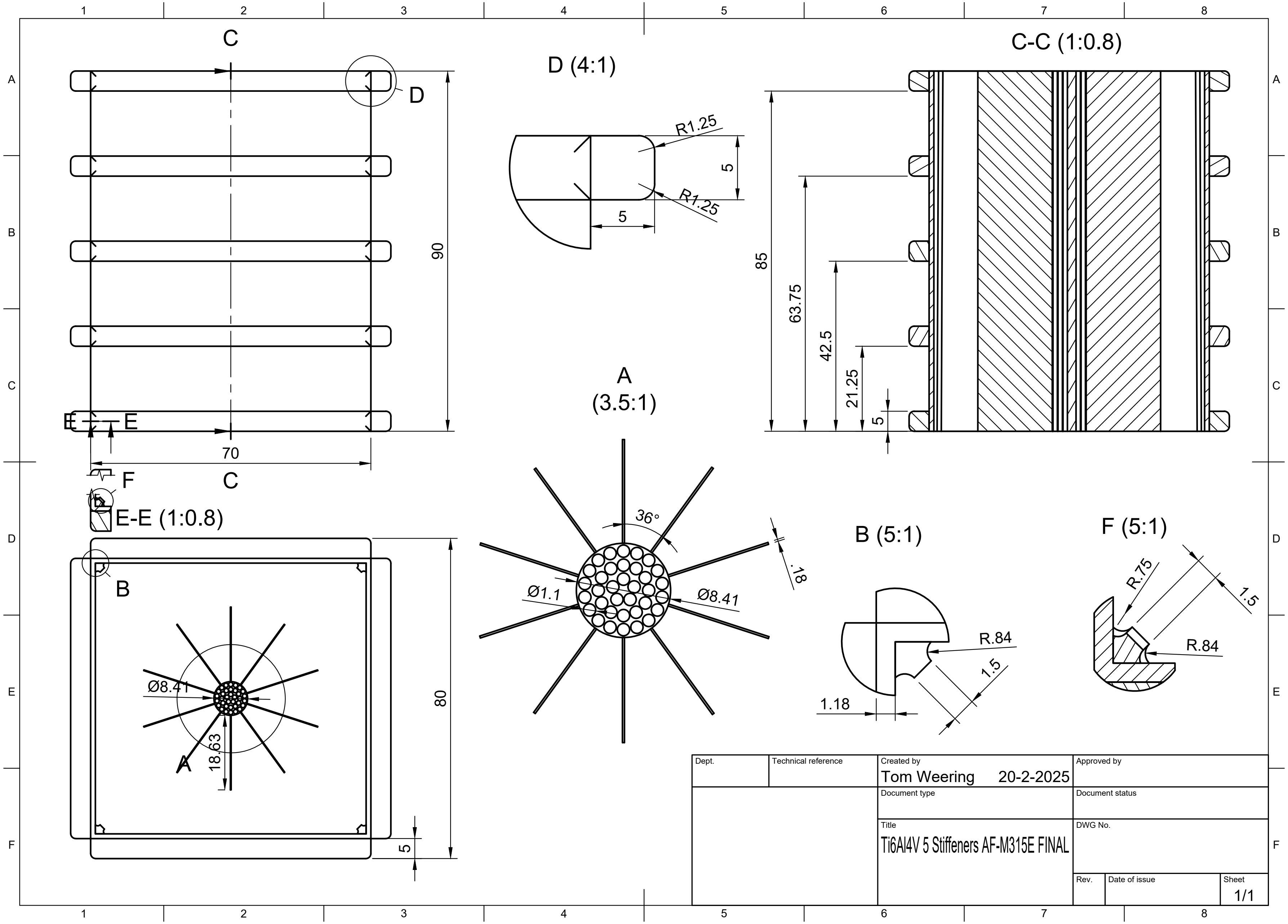


Figure 11: Technical drawing of the selected propellant tank design

References

[1] Hassan Mohamed Abdelalim Abdalla, Daniele Casagrande, Francesco De Bona, Thomas De Monte, Marco Sortino, and Giovanni Totis. An optimized pressure vessel obtained by metal additive manufacturing: Preliminary results. *International Journal of Pressure Vessels and Piping*, 192:104434, 2021.

[2] American Society of Mechanical Engineers. ASME BPVC 2021 Section II Part D (Metric). ASME Boiler and Pressure Vessel Code, 2021. Available at: `https://dl.gasplus.ir/standard-ha/Standard-ASME/ASME%20BPVC%202021%20Section%20II%20part%20D%20(metric).pdf`.

[3] ASME Boiler and Pressure Vessel Committee. *2017 ASME Boiler & Pressure Vessel Code: Rules for Construction of Pressure Vessels, Division 1*. American Society of Mechanical Engineers (ASME), New York, NY, USA, 2017 edition, July 2017.

[4] A. E. Blach, V. S. Hoa, C. K. Kwok, and A. K. W. Ahmed. Rectangular pressure vessels of finite length. *ASME Journal of Pressure Vessel Technology*, 1990. Accessed on 02/18/2017.

[5] Daniel Cavender. Emerging Low Toxicity "Green" Chemical Propulsion Technologies for SmallSats. NASA Technical Reports Server, September 2020.

[6] A. Cervone. *Micro-Propulsion*. Faculty of Aerospace Engineering, Delft University of Technology, v4.0 edition, June 2022. Lecture Notes AE4S07.

[7] Colibrium Additive. Eb-pbf printers. `https://www.colibriumadditive.com/printers/eb-pbf-printers`, 2025. Accessed: 2025-02-20.

[8] Jr. D. E. Jaekle. Propellant management device conceptual design and analysis: Vanes. In *27th AIAA/SAE/ASME/ASEE Joint Propulsion Conference*, number AIAA-91-2172, Sacramento, CA, June 1991. American Institute of Aeronautics and Astronautics (AIAA), AIAA.

[9] Jr. D. E. Jaekle. Propellant management device conceptual design and analysis: Sponges. In *Proceedings of the 29th Joint Propulsion Conference and Exhibit*, number AIAA-93-1970, Monterey, CA, June 28-30 1993. AIAA/SAE/ASME/ASEE. PMD Technology, Andover, MA.

[10] National Center for Biotechnology Information. Pubchem, 2025. Accessed: 2025-02-25.

[11] Jason W. Hartwig. A detailed historical review of propellant management devices for low gravity propellant acquisition. Technical report, NASA Glenn Research Center, Cleveland, OH, USA, 2023.

[12] G. Huggins, A. Talaksi, E. G. Lightsey, D. Andrews, D. Cavender, C. Diaz, D. McQueen, H. Williams, J. Baker, and M. Kowalkowski. Development of a cubesat-scale green monopropellant propulsion system for nasa's lunar flashlight mission. In *Space Propulsion 2022*, pages 9–13, Estoril, Portugal, May 2022. Georgia Institute of Technology, Atlanta, Georgia, 30313; Marshall Space Flight Center, Huntsville, Alabama, 35811; Jet Propulsion Laboratory, California Institute of Technology.

[13] Makerverse. The laser powder bed fusion (l-pbf) design guide, 2025. Accessed: 2025-02-10.

[14] A. E. S. Nosseir, A. Pasini, S. Igarashi, Y. Matsuura, and A. Cervone. Cubesat green propellants am tanks: Efficient designs, propellant management, and chemical compatibility. In *Space Propulsion 2022*, pages 9–13, Estoril, Portugal, May 2022. Universita di Pisa (UniPi), DICI - Sede di Aerospaziale, Pisa, Italy; HI Aerospace Co., Ltd., 900 Fujiki, Tomioka-shi, Gunma, 370-2398, Japan; Delft University of Technology (TU Delft), Faculty of AE dpt. Space Engineering, Delft, The Netherlands.

[15] Ahmed E. S. Nosseir, Angelo Cervone, and Angelo Pasini. Modular impulsive green monopropellant propulsion system (mimps-g): For cubesats in leo and to the moon. *Aerospace*, 8(6):169, 2021.

[16] Georgy A. Shcheglov and Alexey V. Mordovskii. Design optimization of cubic-shaped pressurant tank for cubesat propulsion system. *Acta Astronautica*, 2024.

[17] Ronald A. Spores, Robert Masse, Scott Kimbrel, and Chris McLean. GPIM AF-M315E Propulsion System. In *50th AIAA/ASME/SAE/ASEE Joint Propulsion Conference & Exhibit*, Cleveland, Ohio, July 2014. American Institute of Aeronautics and Astronautics.

[18] A. Steckel. Integrated propellant storage and feed system: Fy18 engineering research technical investment program. Technical Report TIP-89, Lincoln Laboratory, Massachusetts Institute of Technology, Lexington, Massachusetts, May 2019.

[19] PMD Technology. Pmd technology: Propellant management devices (pmds), n.d. Accessed: 2025-02-21.

[20] V. Tepponen, S. Westman, P. Nyamekye, and I. Poutiainen. Optimized inconel 718 pressure vessel manufactured with laser powder bed fusion. In *19th Nordic Laser Materials Processing Conference (NOLAMP-2023)*, volume 1296, page 012019. IOP Publishing, 2023. Corresponding author: vesa.tepponen@lut.fi.

[21] The CubeSat Program, Cal Poly SLO. Cubesat design specification (cds) rev. 13. Technical report, The CubeSat Program, California Polytechnic State University, San Luis Obispo, 2023. Public Domain.

[22] Wikipedia Contributors. SBUDNIC. `https://en.wikipedia.org/wiki/SBUDNIC`, 2023. Accessed: 2025-02-19.

[23] Ziheng Wu, Sneha Prabha Narra, and Anthony Rollett. Exploring the fabrication limits of thin-wall structures in a laser powder bed fusion process. *The International Journal of Advanced Manufacturing Technology*, 110:191–207, August 2020. Received: 26 March 2020 / Accepted: 23 July 2020 / Published online: 11 August 2020.

Code

The full code base can be found at [github].

main.py

```python
import sys
import os
import toml
import math
import copy
import csv
import numpy as np
import matplotlib.pyplot as plt
import matplotlib.cm as cm
from scipy.interpolate import griddata
from ASME._Appendix13_7_a import Appendix13_7_aParams, Appendix13_7_aCalcs
from ASME._Appendix13_8_e import _Appendix13_8_eCalcs
from ASME._Appendix13_7_c import Appendix13_7_cParams, Appendix13_7_cCalcs
from curve import optimize_stiffener_curve


def read_material_data(filename):
    try:
        with open(filename, "r") as file:
            return toml.load(file)
    except FileNotFoundError:
        print("TOML file not found.")
        return {}


def calculate_thermal_equilibrium(material_properties):
    SOLAR_CONSTANT = 1361  # W/m^2
    BOLTZMANN_CONSTANT = 5.670374419e-8  # W/m^2/K^4
    AREA = 0.1 * 0.1  # m^2 for a 10 cm x 10 cm plate

    absorptivity = material_properties["absorptivity"]
    emissivity = material_properties["emissivity"]

    left_side = absorptivity * AREA * SOLAR_CONSTANT
    right_side_factor = emissivity * BOLTZMANN_CONSTANT * AREA

    temperature_kelvin = (left_side / right_side_factor) ** 0.25
    return temperature_kelvin


def interpolate_allowable_stress(material_properties, temperature_celsius):
    room_temp_stress = material_properties.get("allowable_stress_mpa_room_temp", 0)
    elevated_temp_stress = material_properties.get("allowable_stress_mpa_300C", 0)

    if temperature_celsius <= 20:
        return room_temp_stress
    elif temperature_celsius >= 300:
        return elevated_temp_stress

    interpolated_stress = room_temp_stress + (temperature_celsius - 20) * (
        (elevated_temp_stress - room_temp_stress) / (300 - 20)
    )
    return interpolated_stress


def run_pressure_vessel_analysis(material_properties, allowable_stress):
    thickness_range = np.linspace(0.1, 7.5, 500) / 1000  # Thickness values in meters
    pressure_range = np.linspace(0.01, 2, 100) * 10**6  # Pressure values in Pa

    points = []
    labels = []

    for thickness in thickness_range:
        for pressure in pressure_range:
            params_inner = Appendix13_7_aParams(
                long_side_length_inside=(70 / 1000),
                short_side_length_inside=(70 / 1000),
                internal_pressure=pressure,
                short_side_thickness=thickness,
                long_side_thickness=thickness,
                allowable_stress=allowable_stress,
                joint_efficiency=1,
            )

            params_outer = copy.deepcopy(params_inner)
            params_outer.evalAtOuterWalls = True

            calc_inner = Appendix13_7_aCalcs(params_inner)
            calc_outer = Appendix13_7_aCalcs(params_outer)

            max_inner_stress = max(
                abs(calc_inner.S_T_N()), abs(calc_inner.S_T_Q_short()), abs(calc_inner.S_T_M()), abs(calc_inner.S_T_Q_long())
            )
            max_outer_stress = max(
                abs(calc_outer.S_T_N()), abs(calc_outer.S_T_Q_short()), abs(calc_outer.S_T_M()), abs(calc_outer.S_T_Q_long())
            )

            stress_limit = 1.5 * allowable_stress

            is_safe = max_inner_stress <= stress_limit and max_outer_stress <= stress_limit

            points.append((thickness * 1000, pressure / 10**6))  # Convert thickness to mm, pressure to MPa
            labels.append(is_safe)

    return points, labels

def run_rounded_structure_analysis(material_properties, allowable_stress):
    thickness_range = np.linspace(0.1, 7.5, 500) / 1000  # Thickness values in meters
    pressure_range = np.linspace(0.01, 2, 100) * 10**6  # Pressure values in Pa

    points = []
    labels = []

    for thickness in thickness_range:
        for pressure in pressure_range:
            # Define input parameters for Appendix 13-7(c) analysis
            params_inner = Appendix13_7_cParams(
                internal_pressure=pressure,
                corner_radius= 100 / 1000,  # 15 mm in meters
                short_side_half_length=70 / 2 / 1000,  # 35 mm as half length in meters
                long_side_half_length=70 / 2 / 1000,  # 35 mm as half length in meters
                thickness=thickness
            )

            calc_inner = Appendix13_7_cCalcs(params_inner)

            # Evaluate at outer walls
            params_outer = copy.deepcopy(params_inner)
            params_outer.eval_at_outer_walls = True
            calc_outer = Appendix13_7_cCalcs(params_outer)

            # Maximum stress evaluations
            max_inner_stress = max(
                abs(calc_inner.S_T_C()), abs(calc_inner.S_T_D()), abs(calc_inner.S_T_A()), abs(calc_inner.S_T_B())
            )
            max_outer_stress = max(
                abs(calc_outer.S_T_C()), abs(calc_outer.S_T_D()), abs(calc_outer.S_T_A()), abs(calc_outer.S_T_B())
            )

            stress_limit = 1.5 * allowable_stress

            is_safe = max_inner_stress <= stress_limit and max_outer_stress <= stress_limit

            points.append((thickness * 1000, pressure / 10**6))  # Convert thickness to mm, pressure to MPa
            labels.append(is_safe)

    return points, labels


def run_stiffened_structure_analysis(material_properties, allowable_stress, n_stiffeners):
    thickness_range = np.linspace(0.1, 7.5, 500) / 1000  # Thickness values in meters
    pressure_range = np.linspace(0.01, 2, 100) * 10**6  # Pressure values in Pa

    points = []
```

```python
    labels = []

    side_length = 70 / 1000
    stiffner_thickness = 5 / 1000
    stiffner_height = 5 / 1000
    total_height = 90 / 1000
    #n_stiffeners = 5

    pitch = (total_height - stiffner_height) / (n_stiffeners - 1)

    for thickness in thickness_range:
        for pressure in pressure_range:
            params = {
                "P": pressure,
                "H": 0.07,
                "h": 0.07,
                "t_1": thickness,
                "t_2": thickness,
                "ts_1": stiffner_thickness,
                "ts_2": stiffner_thickness,
                "A_1": 0.07 * stiffner_thickness,
                "A_2": 0.07 * stiffner_thickness,
                "H_1": stiffner_height,
                "h_1": stiffner_height,
                "p": pitch,
                "S": allowable_stress,
                "S_y": material_properties.get("yield_strength_mpa", 0)*10**6,
                "E_2": material_properties.get("modulus_of_elasticity_gpa", 0) *10**9,
                "E_3": material_properties.get("modulus_of_elasticity_gpa", 0) *10**9,
            }

            params_obj = type("Params", (object,), params)()
            calc = _Appendix13_8_eCalcs(params_obj)

            max_stress = max(
                abs(calc.S_T_N()), abs(calc.S_T_Q_short()), abs(calc.S_T_M()), abs(calc.S_T_Q_long())
            )

            stress_limit = 1.5 * allowable_stress

            is_safe = max_stress <= stress_limit

            points.append((thickness * 1000, pressure / 10**6))  # Convert thickness to mm, pressure to MPa
            labels.append(is_safe)

    return points, labels



def plot_pressure_thickness_analysis(materials_data):
    # Ask the user which structure type to plot
    structure_type = input("Enter ’unstiffened’ to plot the unstiffened structure, ’stiffened’ to plot the stiffened structure, or ’rounded’
to plot the rounded structure: ").strip().lower()

    # Create the figures directory if it doesn’t exist
    figures_dir = os.path.join(os.path.dirname(os.path.abspath(__file__)), "figures")
    if not os.path.exists(figures_dir):
        os.makedirs(figures_dir)

    fig, axes = plt.subplots(2, 2, figsize=(14, 10))
    plt.subplots_adjust(hspace=0.4, wspace=0.4)

    for ax, (material_name, material_properties) in zip(axes.ravel(), materials_data.items()):
        thermal_equilibrium_temp = calculate_thermal_equilibrium(material_properties)
        allowable_stress = interpolate_allowable_stress(material_properties, thermal_equilibrium_temp)
        allowable_stress = allowable_stress * 10**6  # Convert to Pa

        if structure_type == ’stiffened’:
            points, labels = run_stiffened_structure_analysis(material_properties, allowable_stress, n_stiffeners=5)
        elif structure_type == ’rounded’:
            points, labels = run_rounded_structure_analysis(material_properties, allowable_stress)
        else:
            points, labels = run_pressure_vessel_analysis(material_properties, allowable_stress)

        points = np.array(points)
        labels = np.array(labels)

        # Prepare grid for contour plot
        thickness = points[:, 0]
        pressure = points[:, 1]
        safe_zone = labels.astype(int)

        grid_thickness, grid_pressure = np.meshgrid(
            np.linspace(thickness.min(), thickness.max(), 200),
            np.linspace(pressure.min(), pressure.max(), 200)
        )

        grid_points = np.column_stack((grid_thickness.ravel(), grid_pressure.ravel()))
        interpolated_labels = griddata(points, safe_zone, grid_points, method="linear")

        # Plot hatched regions
        ax.contourf(
            grid_thickness, grid_pressure, interpolated_labels.reshape(grid_thickness.shape),
            levels=[0, 0.5, 1], colors=["red", "green"], alpha=0.3
        )

        ax.set_title(f"Material: {material_name}")
        ax.set_xlabel("Thickness (mm)")
        ax.set_ylabel("Pressure (MPa)")
        ax.set_xlim(thickness.min(), thickness.max())
```

```python
        ax.set_ylim(pressure.min(), pressure.max())
        ax.grid(True)

    plt.suptitle(f"{structure_type} Pressure Vessel Safety Analysis for Different Materials")

    # Save the plot to the figures directory
    plot_path = os.path.join(figures_dir, f"pressure_thickness_analysis_{structure_type}.png")
    plt.savefig(plot_path)
    plt.show()


def extract_feasible_solution(points, labels, target_pressure=2):
    """
    Extract the first feasible solution for a pressure near 2 MPa
    """
    for idx, point in enumerate(points):
        pressure = point[1]  # Pressure in MPa
        if abs(pressure - target_pressure) < 0.1 and labels[idx]:  # Check near 2 MPa and safe design
            return point[0]  # Return thickness (mm)
    return None


def calculate_mass_cost(material_properties, thickness, n_stiffeners):
    """
    Calculate mass and cost of the design based on thickness and number of stiffeners
    """
    side_length = 70 / 1000
    height = 90 / 1000
    stiffener_thickness = 5 / 1000
    stiffener_height = 5 / 1000
    design_volume = (2 * side_length * thickness + (side_length - 2 * thickness) * thickness) * height

    if n_stiffeners > 0:
        stiffener_volume = 4 * n_stiffeners * side_length * stiffener_thickness * stiffener_height
        design_volume += stiffener_volume

    density = material_properties.get("density_kg_m3", 0)
    price_per_kg = material_properties.get("price_per_kg_eur", 0)

    mass = design_volume * density
    cost = mass * price_per_kg

    return mass, cost


def plot_mass_vs_cost(materials_data):
    """
    Plot mass vs. cost for different materials and stiffener configurations
    """
    fig, ax = plt.subplots(figsize=(10, 6))

    markers = {
        "Unstiffened": "o",
        "2 Stiffeners": "s",
        "3 Stiffeners": "^",
        "4 Stiffeners": "v",
        "5 Stiffeners": "D"
    }

    all_points = []
    material_colors = {}
    color_map = cm.get_cmap(’tab10’, len(materials_data))

    for idx, (material_name, material_properties) in enumerate(materials_data.items()):
        thermal_equilibrium_temp = calculate_thermal_equilibrium(material_properties)
        allowable_stress = interpolate_allowable_stress(material_properties, thermal_equilibrium_temp)
        allowable_stress *= 10 ** 6

        masses = []
        costs = []
        labels = []

        # Unstiffened structure
        points, safe_labels = run_pressure_vessel_analysis(material_properties, allowable_stress)
        points = np.array(points)
        safe_labels = np.array(safe_labels)

        thickness = extract_feasible_solution(points, safe_labels)
        if thickness:
            thickness_m = thickness / 1000
            mass, cost = calculate_mass_cost(material_properties, thickness_m, n_stiffeners=0)
            masses.append(mass)
            costs.append(cost)
            labels.append("Unstiffened")
            all_points.append((mass, cost, material_name, "Unstiffened", thickness))

        # Stiffened structures
        for n_stiffeners in range(2, 6):
            points, safe_labels = run_stiffened_structure_analysis(material_properties, allowable_stress, n_stiffeners)
            points = np.array(points)
            safe_labels = np.array(safe_labels)

            thickness = extract_feasible_solution(points, safe_labels)
            if thickness:
                thickness_m = thickness / 1000
                mass, cost = calculate_mass_cost(material_properties, thickness_m, n_stiffeners=n_stiffeners)
                masses.append(mass)
                costs.append(cost)
                labels.append(f"{n_stiffeners} Stiffeners")
                all_points.append((mass, cost, material_name, f"{n_stiffeners} Stiffeners", thickness))

        # Assign color to the material
        material_colors[material_name] = color_map(idx)

        # Plot mass vs. cost for each material
        for i, label in enumerate(labels):
            ax.scatter(costs[i], masses[i], color=material_colors[material_name], marker=markers[label])
            #ax.annotate(label, (costs[i], masses[i]))

    # Calculate distances from the origin and print the top 10 closest points
    distances = [((mass**2 + cost**2)**0.5, mass, cost, material_name, structure_type, thickness) for mass, cost, material_name, structure_type, thickness in all_points]
    distances.sort()
    print("Top 20 closest points to the origin (0,0):")
    for i in range(20):
        distance, mass, cost, material_name, structure_type, thickness = distances[i]
        print(f"Material: {material_name}, Structure: {structure_type}, Mass: {mass:.2f} kg, Cost: {cost:.2f}   , Thickness: {thickness:.2f} mm")

    ax.set_title("Mass vs Cost for Different Structures and Materials")
    ax.set_xlabel("Cost (  )")
    ax.set_ylabel("Mass (kg)")
    ax.grid(True)

    # Create custom legend
    handles = [plt.Line2D([0], [0], marker=markers[label], color='w', label=label, markerfacecolor='k', markersize=10) for label in markers]
    handles += [plt.Line2D([0], [0], marker='o', color='w', label=material_name, markerfacecolor=color, markersize=10) for material_name, color in material_colors.items()]
    ax.legend(handles=handles, title="Legend")

    # Save and show plot
    plot_path = os.path.join("figures", "mass_cost_tradeoff.png")
    plt.savefig(plot_path)
    plt.show()

    # Save designs to CSV
    csv_path = os.path.join("data", "designs.csv")
    with open(csv_path, mode='w', newline='') as file:
        writer = csv.writer(file)
        writer.writerow(["Material", "Structure Type", "Mass (kg)", "Cost (  )", "Thickness (mm)"])
        for mass, cost, material_name, structure_type, thickness in all_points:
            writer.writerow([material_name, structure_type, mass, cost, thickness])



def capillary_length(material_properties, thickness):
    surface_tension = material_properties.get("surface_tension_mN_per_m", 0)
    density = material_properties.get("density_g_per_cm3", 0)
    #convert to SI units
    surface_tension = surface_tension * 10**-3
    density = density * 10**3

    #assumption on acceleration on 3U cubesat
    m_assumed = 5.5 #[kg]
    T_assumed = 1.0 #[N]
    a = T_assumed / m_assumed #[m/s^2]

    #calculate capillary length
    lamda = (surface_tension / (density * a))**(1/2)

    #get fillet radius to resist capillary effect
    L =math.sqrt((35 - thickness)**2 + (35 - thickness)**2)
    R = L - math.sqrt((lamda**2)/4)

    return lamda, R

#TODO make function that compares PMD (Propellant Management Device) designs
# fillet on the edges to resist the capillary effect
# vanes to transfer fluid from the corners to the center
# sponge to absorb the fluid

#for the vane get surface area from frustum (1/4)lateral area (1/2 for total)
#source: https://www.calculatorsoup.com/calculators/geometry-solids/conicalfrustum.php

def vane_sizing(material_properties):
    flow_rate = 30 #[mL / min] (from MiMPS-G e-micro pump)
    #flow_rate = 420 #[mL / min] (exaggerated)
    viscosity = material_properties.get("absolute_viscosity_mPas", 0)
    density = material_properties.get("density_g_per_cm3", 0)
    surface_tension = material_properties.get("surface_tension_mN_per_m", 0)
    delta = (1.5/1000)  #[m] thickness of "sheet metal vane" (in accordance with printable thickness)
    L = (90/1000)   #[m] length of the vane

    #convert to SI units
    flow_rate = flow_rate * 10**-6 / 60          #[m^3 / s]
    viscosity = viscosity * 10**-3               #[Pa s]
    density = density * 10**3                    #[kg / m^3]
    surface_tension = surface_tension * 10**-3   #[N / m]

    #Slant height of a conical frustum:
    #s =    ((r1 - r2)^2 + h^2)
    #Lateral surface area of a conical frustum:
    # S =    * (r1 + r2) * s =    * (r1 + r2) *    ((r1 - r2)^2 + h^2)
    #Volume of a conical frustum:
    #V = (1/3) *    * h * (r1^2 + r2^2 + (r1 * r2))

    vanes = []

    #Get relation for Rdown and Rup from knowing total height and overhang angle can at most be 45 degrees
    for R in np.arange(delta/2, 10*delta, 0.001):
        for theta in np.arange(0, math.pi/4, 0.001):
            R_down = R
            R_up = R + L * math.tan(theta)

            s = math.sqrt((R_up - R_down)**2 + L**2)
            S = math.pi * (R_up + R_down) * s
            V = (1/3) * math.pi * L * (R_up**2 + R_down**2 + (R_up * R_down))
            A = S / 2                               #[m^2] (1/2 of the TOTAL lateral area)
            #Q = flow_rate / 4                       #assume 4 vanes, one for each corner
            Q = flow_rate


            viscous_losses = (2*viscosity*Q*L)/A    #assume wetted area = surface area
            dynamic_losses = (density*Q**2)/(2*A**2)

            young_laplace = surface_tension * (1/R_down - 1/R_up)

            V_vane = R_up*delta*L + L*(R_up+ delta/2)**2  #assumption on volume of vane

            #Print pressures
            #print(f"viscous_losses: {viscous_losses}, dynamic_losses: {dynamic_losses}, young_laplace: {young_laplace}")

            #Check if absolute value of yound_laplace is greater than combined absolute value of viscous and dynamic losses
            if abs(young_laplace) > abs(viscous_losses + dynamic_losses):
                vane = [R_down, R_up, V_vane]
                vanes.append(vane)


    #Sort vanes based on V_vane and return the smallest one
    min_vane = min(vanes, key=lambda x: x[2])
    #print(f"R_down: {min_vane[0]*1000} mm, R_up: {min_vane[1]*1000} mm")
    #print(f"viscous_losses: {viscous_losses}, dynamic_losses: {dynamic_losses}, young_laplace: {young_laplace}")

    return min_vane

def sponge_sizing(propellant_properties, thickness):
    # sigma, rho, a, V_req, r_sponge, t=0.3e-3, g_min_fab=0.5e-3, g_max_ratio=2
    propellant_name = propellant_properties.get("name", "")

    # Obtain propellant properties
    sigma = propellant_properties.get("surface_tension_mN_per_m", 0) * 10**-3
    rho = propellant_properties.get("density_g_per_cm3", 0) * 10**3
    # assumption on acceleration on 3U cubesat
```

```python
    m_assumed = 5.5 #[kg]
    T_assumed = 1.0 #[N]
    a = T_assumed / m_assumed #[m/s^2]

    # Get V_req through burntime and flowrate
    burntime = 10 #[s] (estimate)
    flowrate = 30 #[mL / min] (from MiMPS-G e-micro pump)
    V_req = (flowrate * 10**-6 / 60) * burntime #[m^3]

    P_bubble_point = 200   # Bubble point pressure (Pa) (assumption)
    #ratio = 47438           # Get thickness through simple cantilever model s source: https://apps.dtic.mil/sti/pdfs/AD1098357.pdf

    sponges = []

    # Cycle through a range of radii and thicknesses
    for N in range(10,36):
        r = (2* sigma)/P_bubble_point
        g_max = math.sqrt(sigma/(a*rho))
        R = (g_max*N)/(2*math.pi)
        h = V_req/(math.pi*r**2)

        #TODO Add requirement to get thickness by setting constraint on percentage of total volume dedicated to PMD
        V_tank = ((70/1000)-2*thickness)**2 * (90/1000 - 2*thickness)
        V_PMD = 0.1 * V_tank
        t = V_PMD / (N*(R-r))

        if h > 90e-3:
            h = 90e-3
            r = math.sqrt((V_req)/(h*math.pi))
            r_lattice = (2* sigma)/P_bubble_point


        if R < 0.05:
            sponge = [propellant_name, r, R, h, t, N, V_PMD, r_lattice]
            sponges.append(sponge)

    # Sort sponges based on V and return the smallest one
    best_design = min(sponges, key=lambda x: x[5])

    if best_design[2] >= 90e-3:
        print("r too large to rely on surface tension or capillary action to prevent gas injection. Lattice is required.")

    return best_design


def PMD_design():
    # Material densities
    densities = {"Ti6Al4V": 4420, "StainlessSteel316L": 8000, "StainlessSteel304L": 7930, "Inconel625": 8440}
    structure_types = {"Unstiffened": 0, "2 Stiffeners": 2, "3 Stiffeners": 3, "4 Stiffeners": 4, "5 Stiffeners": 5}

    designs_file_path = os.path.join("data", "designs.csv")
    output_file_path = os.path.join("data", "PMD_designs.csv")

    infill = 0.15  # assumed for support and non-structural 15% infill

    with open(designs_file_path, mode='r') as file:
        reader = csv.DictReader(file)
        designs = list(reader)

    with open(output_file_path, mode='w', newline='') as file:
        writer = csv.writer(file)
        writer.writerow(["Material", "Structure Type", "Propellant", "Thickness (mm)", "R_down (mm)", "R_up (mm)", "V_vane (m^3)", "Vane Mass (kg)", "r_sponge (mm)", "R_sponge (mm)", "h_sponge (mm)", "t_sponge (mm)", "N_sponge", "V_sponge (m^3)", "r_lattice (mm)", "Sponge Mass (kg)", "Support Mass (kg)", "Support Cost (  )", "Total Mass (kg)", "Total Cost (  )"])

        for design in designs:
            material = design["Material"]
            cost = float(design["Cost (  )"])
            mass = float(design["Mass (kg)"])
            price_per_kg = cost / mass
            structure_type = design["Structure Type"]
            thickness = float(design["Thickness (mm)"]) / 1000  # Convert thickness to meters
            density_material = densities.get(material, 0)

            propellant_data = read_material_data(os.path.join(base_dir, "data", "propellants.toml"))
            if propellant_data:
                propellants = propellant_data.get("propellants", {})
                for propellant_name, propellant_properties in propellants.items():

                    lamda, R = capillary_length(propellant_properties, thickness)
                    lamda = lamda / 1000  # Convert from mm to m
                    R = R / 1000  # Convert from mm to m

                    h = 90 / 1000  # Height in meters

                    # Calculate the volume of the fillet
                    volume_fillet = 0.5 * R * h

                    # Calculate the mass of the fillet assuming 15% infill
                    mass_fillet = volume_fillet * density_material * infill

                    # Get mass of support for stiffeners
                    if structure_type == "Unstiffened":
                        n_stiffeners = 0
                        mass_support = 0
                    else:
                        n_stiffeners = structure_types.get(structure_type, 0)
                        area_support = optimize_stiffener_curve(L=90, h=5, n=n_stiffeners, t=5, plot=False)[1]
                        volume_support = 4 * (n_stiffeners - 1) * 70 * area_support
                        volume_support = volume_support / 10**9  # Convert from mm^3 to m^3
                        mass_support = volume_support * density_material * infill

                    vane_design = vane_sizing(propellant_properties)
                    vane_mass = vane_design[2] * density_material

                    sponge_design = sponge_sizing(propellant_properties, thickness)
                    sponge_mass = sponge_design[6] * density_material

                    total_mass = vane_mass + sponge_mass + mass
                    support_cost = mass_support * price_per_kg
                    total_cost = total_mass * price_per_kg + support_cost

                    #TODO limit the accuracy to 0.00 i.e. 3 significant figures
                    writer.writerow([material, structure_type, propellant_name, thickness * 1000, vane_design[0] * 1000, vane_design[1] * 1000, vane_design[2], vane_mass, sponge_design[1] * 1000, sponge_design[2] * 1000, sponge_design[3] * 1000, sponge_design[4] * 1000, sponge_design[5], sponge_design[6], sponge_design[7] * 1000, sponge_mass, mass_support, support_cost, total_mass, total_cost])

                    #print(f"Material: {material}, Structure Type: {structure_type}, Propellant: {propellant_name}, Thickness: {thickness * 1000} mm, R_down: {vane_design[0] * 1000} mm, R_up: {vane_design[1] * 1000} mm, V_vane: {vane_design[2]} m^3, Vane Mass: {vane_mass} kg, r_sponge: {sponge_design[1] * 1000} mm, R_sponge: {sponge_design[2] * 1000} mm, h_sponge: {sponge_design[3] * 1000} mm, t_sponge: {sponge_design[4] * 1000} mm, N_sponge: {sponge_design[5]}, V_sponge: {sponge_design[6]} m^3, r_lattice: {sponge_design[7] * 1000} mm, Sponge Mass: {sponge_mass} kg, Support Mass: {mass_support} kg, Support Cost: {support_cost}   , Total Mass: {total_mass} kg, Total Cost: {total_cost}   ")

# Normalize using min-max scaling, with an option to invert values for minimization
def normalize(values, invert=False):
    min_val, max_val = min(values), max(values)
    norm_values = [(v - min_val) / (max_val - min_val) if max_val > min_val else 0 for v in values]

    return [1 - v if invert else v for v in norm_values]  # Invert if we want to minimize

def trade_off():
    # Dictionary of volumetric ISPs (gs/cm )
    volumetric_specific_impulse_gs_per_cm3 = {
        "AF_M315E": 391, "LMP_103S": 312.48, "HNP225": 245, "FLP_106": 344.6
    }

    # Read CSV file
    designs_file_path = os.path.join("data", "PMD_designs.csv")
    with open(designs_file_path, mode='r') as file:
```

```python
        reader = csv.DictReader(file)
        designs = list(reader)

    # Extract values for normalization
    mass_values = []
    eff_tank_values = []
    cost_values = []
    r_lattice_values = []
    t_sponge_values = []
    support_mass_values = []

    for design in designs:
        mass_values.append(float(design["Total Mass (kg)"]))  # Want LESS
        cost_values.append(float(design["Total Cost (   )"]))  # Want LESS
        r_lattice_values.append(float(design["r_lattice (mm)"]))  # Want MORE
        t_sponge_values.append(float(design["t_sponge (mm)"]))  # Want MORE
        support_mass_values.append(float(design["Support Mass (kg)"]))  # Want LESS

        thickness = float(design["Thickness (mm)"])
        V_vane = float(design["V_vane (m^3)"])
        V_sponge = float(design["V_sponge (m^3)"])

        V_tank_mm3 = (70 - 2*thickness) ** 2 * (90 - 2*thickness)  # mm
        V_tank = V_tank_mm3 / 10**9  # Convert to m
        V_propellant = (V_tank - V_vane - V_sponge) * (85/90)  # m
        V_propellant_cm3 = V_propellant * 10**6  # Convert to cm
        eff_tank_values.append(V_propellant)  # Want MORE

    # Normalize values (inverting where needed)
    mass_norm = normalize(mass_values, invert=True)  # LESS is better
    eff_tank_norm = normalize(eff_tank_values)  # MORE is better
    cost_norm = normalize(cost_values, invert=True)  # LESS is better
    r_lattice_norm = normalize(r_lattice_values)  # MORE is better
    t_sponge_norm = normalize(t_sponge_values)  # MORE is better
    support_mass_norm = normalize(support_mass_values, invert=True)  # LESS is better

    # Compute printability (favoring higher r_lattice, t_sponge, and lower support mass)
    printability_values = [
        (r + t + s) / 3 for r, t, s in zip(r_lattice_norm, t_sponge_norm, support_mass_norm)
    ]

    # Normalize printability
    printability_norm = normalize(printability_values)

    # Compute final trade-off score using weights
    weights = {"Mass": 4, "Effective Tank Volume": 1, "Cost": 3, "Printability": 2}

    trade_off_scores = [
        (weights["Mass"] * m +
         weights["Effective Tank Volume"] * e +
         weights["Cost"] * c +
         weights["Printability"] * p) / sum(weights.values())
        for m, e, c, p in zip(mass_norm, eff_tank_norm, cost_norm, printability_norm)
    ]

    # Store normalized results
    for i, design in enumerate(designs):
        design["Normalized Mass"] = mass_norm[i]
        design["Normalized Effective Tank Volume"] = eff_tank_norm[i]
        design["Normalized Cost"] = cost_norm[i]
        design["Normalized Printability"] = printability_norm[i]
        design["Trade-off Score"] = trade_off_scores[i]

    # Write updated data with normalized values to a new CSV
    output_file_path = os.path.join("data", "PMD_designs_with_tradeoff.csv")
    with open(output_file_path, mode='w', newline='') as file:
        fieldnames = designs[0].keys()
        writer = csv.DictWriter(file, fieldnames=fieldnames)
        writer.writeheader()
        writer.writerows(designs)

    print(f"Trade-off analysis completed. Results saved to {output_file_path}")

    # Print top 10 designs with normalized values
    designs.sort(key=lambda x: x["Trade-off Score"], reverse=True)
    print("\nTop 10 designs with Normalized Values:")
    for i in range(min(10, len(designs))):
        print(f"""
        Material: {designs[i]['Material']}
        Structure Type: {designs[i]['Structure Type']}
        Propellant: {designs[i]['Propellant']}
        Normalized Mass: {designs[i]['Normalized Mass']:.3f}
        Normalized Effective Tank Volume: {designs[i]['Normalized Effective Tank Volume']:.3f}
        Normalized Cost: {designs[i]['Normalized Cost']:.3f}
        Normalized Printability: {designs[i]['Normalized Printability']:.3f}
        Trade-off Score: {designs[i]['Trade-off Score']:.3f}
        """)

if __name__ == "__main__":
    base_dir = os.path.dirname(os.path.abspath(__file__))
    materials_file_path = os.path.join(base_dir, "data", "materials.toml")

    material_data = read_material_data(materials_file_path)

    if material_data:
        materials = material_data.get("materials", {})
        #plot_mass_vs_cost(materials)

        #plot_pressure_thickness_analysis(materials)
        #plot_stiffened_vs_unstiffened(materials)
```

```python

    PMD_design()
    trade_off()
```

Listing 1: Main script

curve.py

```python
import numpy as np
import matplotlib.pyplot as plt
import scipy.integrate as spi

def optimize_stiffener_curve(L, h, n, t, plot=True):
    """
    Optimize the stiffener support curve while ensuring the slope does not exceed 45 degrees.

    Parameters:
        L (float): Total height of the box.
        h (float): Height of stiffener.
        n (int): Number of stiffeners.
        t (float): Extremity thickness of stiffeners.
        plot (bool): Whether to generate the plot (default: True).

    Returns:
        tuple: (t - y0, area_above_curve)
    """
    # Calculate pitch and curve width
    P = (L - h) / (n - 1)
    curve_width = (P - h) / 2  # Half-width of the curve

    # Start with initial height guess and reduce if needed
    y0 = t

    # Define x range
    x_range = np.linspace(-curve_width, curve_width, 1000)

    # Adjust y0 until all slopes are     45 degrees
    while True:
        # Define the parabolic curve: y = a*x^2 + y0
        a = -y0 / curve_width**2  # Ensuring parabola peaks at (0, y0)
        y_curve = a * x_range**2 + y0

        # Analytical derivative dy/dx = 2 * a * x
        derivative = 2 * a * x_range

        # Compute angles in degrees
        angles = np.abs(np.arctan(derivative) * (180 / np.pi))

        # If all angles are <= 45 degrees, break loop
        if np.all(angles <= 45):
            break

        # Reduce y0 slightly if any angle exceeds 45 degrees
        y0 -= 0.01

    # Bounding box area
    box_area = 2 * curve_width * y0

    # Integrate the area under the curve
    area_under_curve, _ = spi.quad(lambda x: a * x**2 + y0, -curve_width, curve_width)

    # Area above the curve
    area_above_curve = box_area - area_under_curve



    # Generate plot if requested
    if plot:
        plt.figure(figsize=(6, 4))
        plt.plot(x_range, y_curve, label=f"Optimized Curve (y0={y0:.2f})", color="blue")
        plt.axhline(0, color='gray', linestyle='--', label="Stiffener Base")
        plt.axvline(-curve_width, color='red', linestyle='--', label="Extremity")
        plt.axvline(curve_width, color='red', linestyle='--')
        plt.fill_between(x_range, y_curve, y0, color='lightblue', alpha=0.5, label="Area Above Curve")
        plt.xlabel("Width (x)")
        plt.ylabel("Height (y)")
        plt.legend()
        plt.title("Optimized Support Curve with Area Calculation")
        plt.grid()
        plt.show()

        # Print results
        print(f"Optimized curve height (y0) to keep slope    45  : {y0:.4f}")
        print(f"Adjusted thickness (t - y0): {t - y0:.4f}")
        print(f"Area above the curve: {area_above_curve:.4f} [mm^2]")

    return t - y0, area_above_curve

# Example Usage
#adjusted_thickness, area_above = optimize_stiffener_curve(L=90, h=5, n=2, t=5, plot=True)
```

Listing 2: Optimizes curve for support structure