Stress-Guided Material Segmentation for Recycled 3D Printed Structures Using Finite Element Analysis

by

Cole J. Paulin

B.S. Electrical Engineering and Computer Science, MIT, 2024

Submitted to the Department of Electrical Engineering and Computer Science
in partial fulfillment of the requirements for the degree of

MASTER OF ENGINEERING IN ELECTRICAL ENGINEERING AND COMPUTER SCIENCE

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

May 2025

Authored by: Cole J. Paulin
Department of Electrical Engineering and Computer Science
May 9, 2025

Certified by: Stefanie Mueller
Associate Professor, Thesis Supervisor

Accepted by: Katrina LaCurtis
Chair, Master of Engineering Thesis Committee

Stress-Guided Material Segmentation for Recycled 3D Printed Structures Using Finite Element Analysis

by

Cole J. Paulin

Submitted to the Department of Electrical Engineering and Computer Science
on May 9, 2025 in partial fulfillment of the requirements for the degree of

MASTER OF ENGINEERING IN ELECTRICAL ENGINEERING AND COMPUTER SCIENCE

ABSTRACT

We present a simulation-driven method for optimizing the structural performance of 3D printed objects made with recycled and fresh filament. Although sustainable materials such as recycled PLA reduce environmental impact, they often exhibit degraded or inconsistent mechanical properties, making them less suitable for structurally demanding applications. To address this, we develop a finite element analysis (FEA) pipeline that simulates stress and strain distributions under user-defined loading conditions, enabling intelligent segmentation of the object into regions of high and low mechanical demand. These segmented regions can be assigned recycled or fresh material during fabrication. Our system leverages open-source tools (SfePy) for simulation and we validate its accuracy against Abaqus, a commercial industry standard. We also introduce methods for automatically identifying and correcting segmentation artifacts, such as small disconnected islands. Through comparative simulation studies and performance evaluation, we demonstrate that our approach enables more sustainable 3D printing without sacrificing structural reliability.

Thesis supervisor: Stefanie Mueller

Title: Associate Professor

Acknowledgments

This thesis would not have been possible without the support and collaboration of many people.

I am deeply grateful to Maxine Perroni-Scharf for her mentorship and guidance throughout this project. Her leadership helped shape both the direction of the work and the growth of our team. I would also like to thank my teammates, Zhi Ray Wang and Jennifer Xiao, for their creativity, technical insight, and dedication throughout the development of this project.

I am thankful to Professor Stefanie Mueller for bringing me onto the team and for her thoughtful advising and encouragement along the way. I also want to thank MIT for the incredible opportunities, resources, and community that have supported me through both my undergraduate and master's studies, and for helping me grow as a thinker, engineer, and researcher.

Finally, I would like to thank my parents, Mark and Meesha Paulin, for their unwavering love and support. I could not have done any of this without them.

Contents

List of Figures

Chapter 1

Introduction

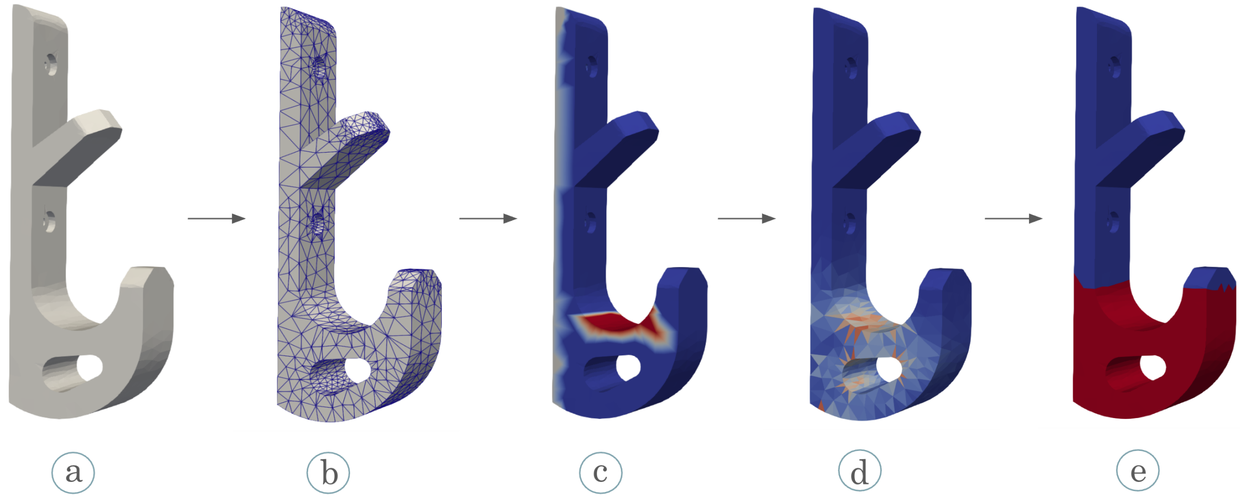


Figure 1.1: Overview of the stress simulation workflow. **(a)** Input geometry. **(b)** Volumetric mesh generated using `fTetWild`. **(c)** User-defined boundary conditions applied to load and fixed regions. **(d)** Output stress field from finite element analysis. **(e)** Segmented mesh indicating regions assigned to recycled (blue) and virgin (red) filament.

1.1 Overview

Personal fabrication research has enabled makers to rapidly produce customized objects through 3D printing. As these capabilities expand, the environmental impact of plastic consumption has drawn increasing attention [1–3]. Recent work has explored sustainable approaches such as recycled composites [4, 5], compostable textiles [6], and repurposing print waste [7, 8]. However, these methods often target waste management, not the core challenge: the mechanical limitations of sustainable materials themselves.

Recycled and biodegradable filaments reduce reliance on petroleum-based polymers, but typically exhibit substantial mechanical degradation—with tensile strength losses of up to 20% after multiple recycling cycles [4, 5, 9]. This performance gap restricts their use in load-bearing prints, discouraging adoption despite environmental benefits.

This thesis presents a simulation-driven pipeline that assists users in strategically assigning recycled and virgin filaments within a single print. The system uses Finite Element Analysis (FEA) to predict internal stresses under expected loading conditions, and automatically segments the mesh into strong and weak material regions based on these stress fields. Recycled filament is assigned by default, while virgin filament is selectively applied where mechanical failure would otherwise occur.

Key contributions include a validated simulation framework using `SfePy`, stress-based segmentation strategies, robust heuristics for eliminating small disconnected material islands, and validation workflows comparing open-source and industry-standard simulation tools. Together, these contributions enable more sustainable 3D printing without compromising structural performance.

1.2 Motivation and Problem Statement

Desktop 3D printing has made personalized manufacturing widely accessible, but its growth has also amplified environmental concerns [2, 10]. Eco-friendly filaments such as recycled PLA offer a path forward, yet suffer from reduced strength and limited mechanical data availability [9, 11, 12]. Makers face a difficult choice: prioritize sustainability at the cost of structural reliability, or revert to petroleum-based virgin materials.

This problem is compounded by the lack of computational tools that allow users to proactively manage material assignment based on structural needs. Current workflows often rely on intuition or manual trial-and-error, resulting in excessive material switching, unpredictable mechanical behavior, or failed prints.

This thesis addresses the gap by introducing a simulation-backed method to systematically guide the allocation of recycled and virgin materials within a print. Specifically, it tackles three core challenges:

- **Predicting high-stress regions:** Using accessible finite element analysis tools to forecast where mechanical reinforcement is necessary.
- **Optimizing material assignment:** Developing segmentation methods that maximize recycled material use without compromising critical load-bearing areas.
- **Ensuring manufacturability:** Eliminating small, disconnected regions ("islands") that could undermine print quality or mechanical integrity.

By combining open-source simulation tools with customized segmentation and post-processing strategies, this work provides makers with an accessible, practical pathway toward reliable sustainable fabrication.

1.3 Background on Computational Tools

Accurate simulation and analysis tools are essential for enabling stress-informed material assignment. This thesis leverages a combination of open-source and industry-standard software packages to simulate, validate, and visualize internal stress distributions.

1.3.1 SfePy (Simple Finite Elements in Python)

SfePy is an open-source Python library for solving partial differential equations (PDEs) using finite element methods [13]. Its flexibility, Python integration, and modularity make it suitable for custom simulation pipelines. In this work, SfePy is used to model stress and strain behavior in 3D-printed objects subjected to user-defined loading conditions.

1.3.2 Abaqus

Abaqus is a commercial finite element analysis software widely used for structural simulation [14]. Known for its reliability in modeling complex mechanical behavior, Abaqus serves here as a validation tool: simulations produced by the open-source pipeline are compared against Abaqus outputs to ensure fidelity.

1.3.3 fTetWild

fTetWild [15] is a robust tetrahedral meshing tool that converts noisy or imperfect surface meshes into high-quality volumetric meshes suitable for simulation. Unlike traditional meshing software, fTetWild is resilient to self-intersections and small sliver elements, ensuring reliable results even with non-ideal user-generated models.

1.3.4 ParaView

ParaView [16] is an open-source scientific visualization platform designed for large-scale simulation datasets. Within this thesis, ParaView is used to visualize simulation results, stress fields, and segmented material regions, enabling intuitive exploration of model behavior.

1.4 Scope and Limitations

This thesis focuses on the computational design and evaluation of sustainable 3D prints using stress-informed material segmentation. The methods are primarily developed and validated for Polylactic Acid (PLA) and recycled PLA filaments, which are common in desktop fused deposition modeling (FDM).

1.4.1 Scope

The system enables users to selectively reinforce regions of a print by combining:

- Finite element stress simulations based on user-defined loading conditions,
- Material segmentation into recycled and virgin filament regions,
- Island removal heuristics to ensure printability and mechanical reliability.

Simulations are performed using SfePy, with validation against Abaqus for benchmarking. The methods are generalizable but are tuned for static load scenarios and the material properties of PLA-based filaments.

1.4.2 Limitations

Several assumptions and constraints apply:

- **Material Models:** Only linear elasticity, Neo-Hookean, and Mooney-Rivlin constitutive models are supported.
- **Loading Conditions:** Static, user-defined load scenarios are assumed; dynamic or fatigue loading is not modeled.
- **Material Behavior:** Materials are assumed homogeneous and isotropic; effects like anisotropy from printing or viscoelasticity are not considered.
- **Mesh Resolution:** Stress accuracy is limited by tetrahedral mesh quality generated by `fTetWild`.
- **Segmentation Parameters:** Thresholds for segmentation and island removal are manually tuned and may affect reproducibility.

Future work could expand this framework to support broader material types, more complex loading scenarios, and automated boundary condition inference.

1.5 Research Objectives

This thesis aims to:

1. Develop a computationally-driven pipeline for selectively assigning recycled and virgin PLA filament based on simulated stress analysis.
2. Validate computational methods for assessing stress distributions through comparisons with industry-standard FEA tools.
3. Demonstrate a practical method for sustainable 3D printing that does not compromise structural integrity.

1.6 Structure of the Thesis

This document is organized as follows:

- **Chapter 2: Related Work** reviews prior work in sustainable 3D printing materials, fabrication-aware design, finite element simulation, and user-friendly fabrication tools.
- **Chapter 3: Methodology** describes the full computational pipeline, including stress simulation using SfePy, material segmentation methods, and island removal techniques.
- **Chapter 4: Applications** presents real-world examples such as hooks, planters, and stands, demonstrating how the pipeline can be used to reinforce functional objects while using mostly recycled filament.
- **Chapter 5: Evaluation** presents the technical validation of the system, including runtime analysis, comparison against Abaqus, and a study of island removal methods. It concludes with a clearly labeled summary of broader mechanical testing conducted by the team.

- **Chapter 6: Conclusion and Future Work** summarizes key contributions, limitations, and opportunities for further improving sustainable fabrication workflows.

1.7 Key Terms and Definitions

- **Finite Element Analysis (FEA)**: A computational technique used to predict how objects behave under various physical conditions by breaking down complex geometries into simpler elements [17].
- **Recycled PLA (Polylactic Acid)**: PLA filament produced from previously used PLA material, typically exhibiting reduced mechanical performance compared to virgin PLA [4, 9].
- **Sustainable Fabrication**: The practice of designing and producing objects with methods and materials intended to minimize negative environmental impacts [2, 3].
- **Material Segmentation**: Computationally determining material assignment regions within a multi-material print based on performance criteria such as predicted stress or strain.

1.8 Expected Contributions

This thesis anticipates making the following contributions:

- A validated computational framework for selective use of recycled and virgin filaments in sustainable 3D printing.
- Methodological advancements in stress-informed filament assignment using accessible simulation software.
- Empirical evidence demonstrating the feasibility and structural benefits of optimized sustainable printing methods.

1.9 Broader Impact

This work supports broader sustainability goals by enabling more reliable use of recycled materials in structurally critical contexts. By reducing the reliance on virgin plastics without compromising mechanical integrity, the methods developed herein contribute toward achieving more sustainable manufacturing practices and help foster environmentally responsible consumer behaviors.

Chapter 2

Related Work

Unlike previous multi-material printing systems aimed at aesthetics or geometry reuse, this project focuses on reinforcing recycled filament prints to maintain strength, directly addressing the gap in sustainable fabrication workflows and connecting fabrication-aware geometry design with the use of sustainable materials. In this section, we give an overview of sustainability in HCI research and fabrication practices and outline how our approach brings these two areas together.

2.1 Sustainable Digital Fabrication in HCI

A growing body of HCI research focuses on making personal fabrication more sustainable [1], by intervening at various stages of the fabrication lifecycle, from raw material acquisition to end-of-life disposal [18]. Researchers have developed tools across the spectrum—from recycling electronic waste into new devices [19], to zero-waste fashion [20], to helping users adopt greener transportation habits [21].

Specifically in 3D printing, a recent survey of makerspaces [22] highlights persistent issues with plastic waste and a lack of tools to address them. Prior research addresses this challenge through two main strategies: optimizing the *materials* used—such as reusing scraps or developing biodegradable filaments—and optimizing the *geometry* of prints to reduce material consumption.

Several systems target material savings through geometry modifications. *WirePrint* [23] prints sparse wireframe previews to save filament, while Jones et al. [24] optimize infill patterns to reduce plastic use without compromising strength. Chen et al.'s *Encore* [8] attaches new prints to existing objects to avoid redundant printing. *Scrappy* [7] repurposes failed prints as internal infill, and *FusePrint* [25] incorporates recycled fragments directly into new prints. Similarly, *TrussFab* [26] connects 3D-printed nodes with used plastic bottles to form large load-bearing structures.

On the materials side, researchers have explored novel eco-filaments such as Rivera et al.'s spent coffee-based filament [27], biodegradable e-textiles like *EcoThreads* [28], and mycelium-based composites [29]. However, many eco-friendly filaments suffer from reduced strength or brittleness. Commercial products like Polymaker PolyTerra and Terrafilium PLA, while compostable, often lack the mechanical properties needed for structural applications. In practice, makers increasingly purchase filaments marketed as recycled, bio-based, or otherwise sustainable, but little research explores how to adapt fabrication workflows to accommodate their limitations.

To address this gap, this system operates at the intersection of geometry and material properties. We ask: given access to eco-friendly filament, how can we use geometry to maximize its use without compromising structural performance? By combining mechanical simulation with multi-material 3D printing, our approach allows users to reinforce structurally critical regions of a print with virgin filament while using sustainable filament elsewhere.

2.2 Sustainable Filaments and Their Challenges

In parallel to HCI efforts, materials researchers have studied the properties and limitations of recycled and composite 3D printing filaments. Overall, studies show that making filaments "greener" often comes with mechanical trade-offs. For example, recycled PLA-produced by melting and re-extruding waste prints—typically exhibits lower tensile and impact strength than virgin PLA [8]. Hasan et al. [9] review these effects, noting consistent drops in polymer molecular weight and strength across recycling cycles. Beltran et al. found that PLA's tensile strength decreased by 21.6% (from 51 MPa to 40 MPa) after six recycling cycles, along with reduced stiffness and impact resistance. Bio-based additives or high recycled content can also introduce brittleness or inconsistent print quality [24].

Researchers have explored ways to mitigate these issues at the material level. For example, [9] explored chemical additives such as chain extenders and peroxides to restore strength to degraded PLA. Other work investigates reinforcing bioplastics with fibers or nanoparticles [30]. However, these solutions are often inaccessible to everyday makers and do not fully eliminate degradation.

This motivates our approach: rather than requiring specialized materials or chemical knowledge, we provide a fabrication-aware software workflow that compensates for material weaknesses through intelligent design and multi-material printing.

2.3 Fabrication-Aware Design Tools

Another line of prior work improves sustainability by adapting the geometry of fabricated objects or optimizing the printing process. Some approaches reduce material use by printing sparse previews [23], optimizing infill structures [24], or integrating recycled scrap into new designs (*Scrappy* [7], *TrussFab* [26]). Other efforts focus on zero-waste design principles, such as Saakes et al.'s *PacCAM* system for material-efficient furniture design [31].

Beyond sustainability, HCI and graphics researchers have developed fabrication-aware tools that integrate simulation into the design process. Bermano et al. [32] survey such tools, which often use finite element analysis (FEA) or geometric segmentation to improve structural integrity. Recent systems like *Style2Fab* [33] use stress simulations to guide both aesthetic and functional remixing of models.

We build on these ideas by using simulation not just for strength, but to explicitly balance structural integrity with sustainability goals—allowing users to combine virgin and recycled filament in a single print and navigate the tradeoffs between maximizing structural performance and maximizing the use of sustainable material. Unlike prior tools that optimize for aesthetics or geometry reuse, we explicitly target mechanical compensation for degraded materials—enabling sustainable printing without requiring new filament formulations or trial-and-error tuning. Whereas *Style2Fab* segments models for functional aesthetics, we segment models based on simulated stress and assigns materials accordingly—reinforcing high-stress regions with strong filament while preserving weaker, eco-friendly materials elsewhere. This technique integrates material-aware and geometry-aware strategies into a cohesive, fabrication-aware tool that supports greener, stronger 3D printing.

2.4 Simulation-Driven Design and Structural Analysis in Fabrication

While much prior work in sustainable digital fabrication has focused on material reuse, infill optimization, or design heuristics, relatively little attention has been paid to integrating structural simulation into consumer-grade fabrication workflows. However, in engineering and graphics, stress simulation using Finite Element Analysis (FEA) has long been a standard technique for evaluating mechanical performance before fabrication.

Recent systems like Style2Fab [34] and the survey by Bermano et al. [32] illustrate how stress-aware modeling can guide design choices—typically for aesthetic or ergonomic modifications. However, these tools stop short of using simulation to guide material-level decisions. In contrast, this thesis applies FEA to identify structurally critical regions and dynamically assign stronger virgin filament to those areas, while maximizing the use of recycled PLA elsewhere.

To enable this, we leverage SfePy [13], an open-source Python-based FEA library, for simulation, and validate the outputs against Abaqus [14], an industry-standard commercial FEA tool. Additionally, we use fTetWild [15] for robust tetrahedral meshing of arbitrary 3D models and ParaView [16] for cross-platform result visualization.

By incorporating stress-based analysis into the sustainable fabrication workflow, this approach empowers makers to design with both material efficiency and structural safety in mind. It complements prior sustainable systems by grounding material assignment in measurable mechanical metrics, rather than purely visual or empirical heuristics.

2.5 Life Cycle Assessment and Environmental Impact

Recent research in sustainable fabrication emphasizes the evaluation of the environmental impact of 3D printing beyond just material use. Life Cycle Assessment (LCA) offers a quantitative framework for assessing energy, emissions and resource usage throughout the product lifecycle [35]. In the context of additive manufacturing, LCA studies have revealed significant differences in sustainability outcomes depending on filament source, machine settings, and part design [36]. Notably, switching from virgin to recycled PLA can reduce embodied energy and carbon emissions, but must be balanced against performance losses [37].

2.6 Multi-Material and Functionally Graded Printing

Multi-material and functionally graded printing methods enable spatial control over material properties, often used in soft robotics, medical prosthetics, or compliance-optimized parts. Techniques like voxel-based modeling [38] and voxel-level printing [39] give designers fine-grained control over how different materials are distributed. Although these tools typically target aesthetic or functional variation, this system applies similar strategies to sustainability—assigning stronger filament only where mechanically necessary, while allowing the bulk of the print to be composed of recycled material. This novel reuse of functionally graded design principles for eco-conscious fabrication expands the design space of sustainable HCI.

2.7 Fabrication Tools for Non-Experts

The HCI community has made great strides in building fabrication-aware tools for novice users, allowing people with limited engineering expertise to engage in complex digital making. Examples include systems that help remix 3D models [40], embed sensors or electronics [41], and adapt toolpaths to curved surfaces [42]. These tools reduce barriers by abstracting complexity. We aim to add to this field by creating a simple UI and backend, easy for even a novice.

Chapter 3

Methodology

3.1 Overview of the Pipeline

This system is a simulation-based design pipeline that enables makers to optimize the use of sustainable filament in functional 3D printed objects. The system takes as input a 3D model—typically provided in '.obj' or '.stl' format—and produces a segmented mesh with material assignments indicating which regions should be printed using virgin PLA and which can safely use recycled PLA. This segmentation is informed by mechanical simulation under expected loading conditions and aims to reinforce structurally critical regions while maximizing the use of sustainable filament elsewhere.

Figure 3.1 provides a high-level visualization of this pipeline, from simulation to physical fabrication. The steps described below correspond to each stage shown in the figure.

The pipeline consists of the following stages:

1. **Mesh Conversion and Preprocessing:** Input surface meshes are converted into volumetric tetrahedral meshes using `fTetWild` [15]. Users may optionally apply mesh rotations and define spatial bounding boxes for fixing and loading specific regions.

2. **Finite Element Simulation:** The tetrahedral mesh is processed in `SfePy` [13], an open-source Python finite element analysis library. User-defined material parameters,

boundary conditions, and load directions are specified through configuration files. The simulation outputs internal stress and strain distributions based on the defined loading scenario.

3. **Stress-Based Segmentation:** Based on the simulated von Mises stress field, the system applies a segmentation heuristic to distinguish between high-stress and low-stress regions. These are assigned virgin and recycled PLA, respectively. Users can select between threshold-based or layer-based segmentation strategies to control how material regions are defined.

4. **Island Removal and Post-Processing:** To improve printability and mechanical cohesion, small isolated tetrahedral clusters (islands) are detected and reassigned based on neighboring material composition. This ensures that all printable regions form coherent structural regions without fragile outliers.

5. **Output Export and Visualization:** The segmented mesh and simulation results are exported in `.vtu` or `.vtk` format for visualization using ParaView [16]. These outputs can then be used to inform multi-material slicing and fabrication.

While the pipeline itself is built entirely on open-source tools, the accuracy of the simulation was manually verified by comparing SfePy results against equivalent simulations performed in Abaqus [14] on one of the benchmark models. This one-time comparison supports the validity of the stress distributions produced by the pipeline, ensuring that it can be trusted for guiding material assignment in real-world prints.

3.2 Mesh Preparation and Preprocessing

Before conducting any stress simulations, the 3D model must be prepared and converted into a tetrahedral mesh suitable for finite element analysis. This stage is critical, as the quality and resolution of the mesh directly influence the accuracy and stability of the simulation

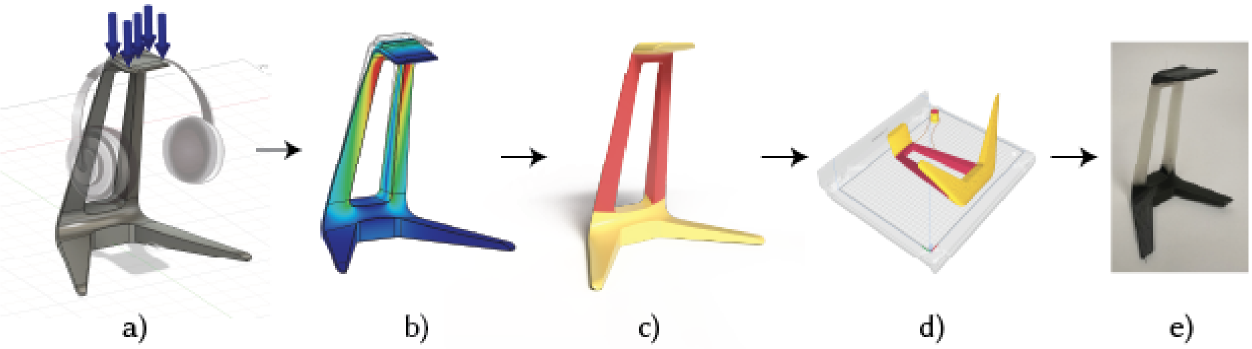


Figure 3.1: Overview of the pipeline: **a)** Apply boundary conditions to the 3D model. **b)** Simulate stress distribution using Finite Element Analysis. **c)** Segment regions into high- and low-stress zones. **d)** Assign materials and slice the model for dual-extrusion printing. **e)** Fabricated part using mixed PLA materials.

results. Figure 3.2 illustrates this entire preprocessing pipeline, from initial model import to the generation of simulation-ready meshes and boundary conditions.

3.2.1 Input Geometry

Users begin by providing a 3D model in a common format such as OBJ or STL. These files are often exported from CAD programs or downloaded from online repositories. To ensure compatibility with simulation tools, the model must be watertight, manifold, and free of non-manifold edges or intersecting geometry.

3.2.2 Tetrahedral Meshing with fTetWild

We use `fTetWild` [15], an open-source tetrahedral meshing library, to convert surface meshes into volumetric tetrahedral meshes. fTetWild is particularly suited to meshing complex, noisy, or imperfect models, which makes it ideal for user-submitted designs.

The tool ensures that the resulting mesh:

- is free of inverted or degenerate elements,
- maintains fidelity to the original surface geometry, and

- is robust to slight imperfections in the input.

The meshing resolution is configurable by the user, with finer meshes yielding more accurate stress fields at the cost of increased computational resources.

3.2.3 Coordinate System and Rotation Handling

To support loading in arbitrary directions (e.g., simulating a device standing upright or laying flat), we allow users to specify a `rotation_vector` in axis-angle format. This rotation is applied to the mesh before simulation to align the model with a canonical coordinate system.

Internally, the mesh is rotated using `scipy`'s `Rotation` class, which performs the transformation efficiently and updates all vertex coordinates accordingly.

3.2.4 Bounding Box Detection for Regions of Interest

The user is prompted to define two axis-aligned bounding boxes:

- A **fixed region**, where displacement constraints will be applied (e.g., the base of a structure),
- A **loading region**, where external forces will be applied.

Vertices within these bounding boxes are automatically identified and grouped into simulation regions. We optionally visualize these selections to allow the user to confirm proper region specification before running the simulation.

3.2.5 Mesh Export for Abaqus Validation

For comparative validation, the tetrahedral mesh is also exported in a format readable by Abaqus. This allows users to simulate the same geometry using both open-source and industry-standard tools.

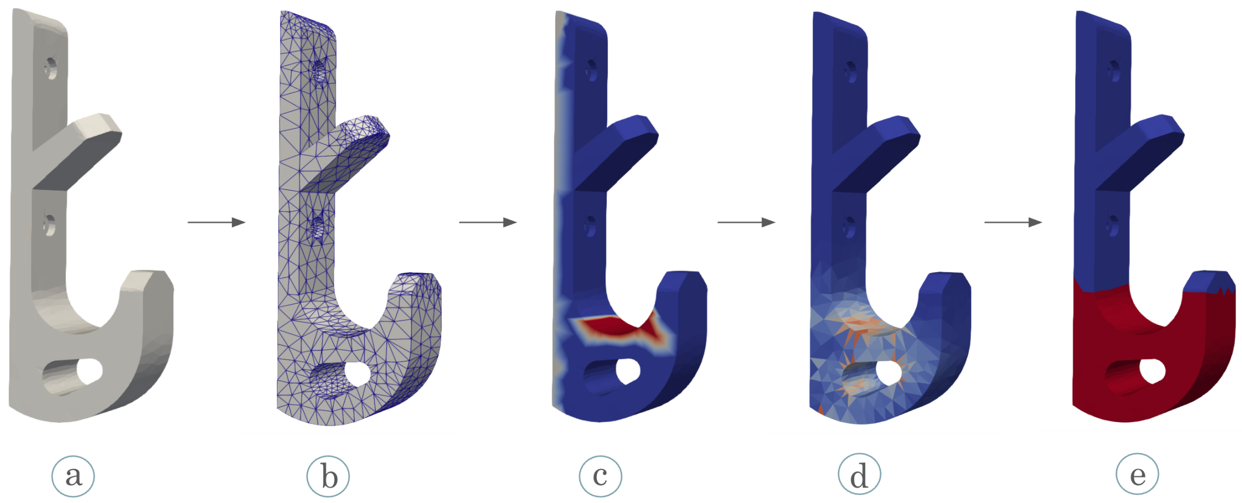


Figure 3.2: Overview of the stress simulation workflow. **(a)** Input geometry. **(b)** Volumetric mesh generated using `fTetWild`. **(c)** User-defined boundary conditions applied to load and fixed regions. **(d)** Output stress field from finite element analysis. **(e)** Segmented mesh indicating regions assigned to recycled (blue) and virgin (red) filament.

3.3 Stress Simulation and Analysis

The stress simulation process continues from this preprocessing step, as shown in Figure 3.2. To assess the structural behavior of each 3D model, we perform finite element stress simulations using the `SfePy` library [13]. These simulations identify regions of high stress under user-defined loading conditions, informing how filament materials are later assigned during segmentation.

3.3.1 Boundary Conditions and Loading

Users configure fixed and loaded regions through axis-aligned bounding boxes specified in `setup_config.yaml`. The fixed region is constrained based on a specified direction (`x`, `y`, `z`, or `all`), and the load region receives a traction force in a custom vector direction. These regions are extracted from the mesh and visualized during preprocessing to ensure correct placement.

Surface tractions are computed per timestep based on the loading direction and magnitude.

The user may choose between tension, compression, or a custom load profile. These options are handled through modular loading functions.

3.3.2 Material Models

Our simulation framework supports three constitutive models:

- **Linear Elasticity:** Assumes small deformations and linearly relates stress to strain through a stiffness tensor derived from Lamé parameters.
- **Neo-Hookean:** A single-invariant hyperelastic model that captures moderate nonlinear behavior typical of polymers like PLA.
- **Mooney-Rivlin:** A more expressive model that incorporates two invariants to handle large deformations and rotations.

These models are implemented using total Lagrangian formulations where applicable, allowing support for large strain regimes when needed.

3.3.3 Simulation Workflow

Each simulation is run over a series of timesteps (e.g., $n\_step = 10$), incrementally applying load and solving for the resulting displacement field.

3.4 Material Segmentation

Once stress magnitudes have been computed through simulation, the next step in the pipeline is to segment the mesh into regions to be printed with different materials. The goal of material segmentation is to preserve structural performance by reinforcing high-stress areas with stronger (virgin) PLA while allowing the remainder of the model to be printed with recycled filament. To support diverse object geometries and use cases, we implemented two segmentation strategies: threshold-based segmentation and layer-based segmentation.

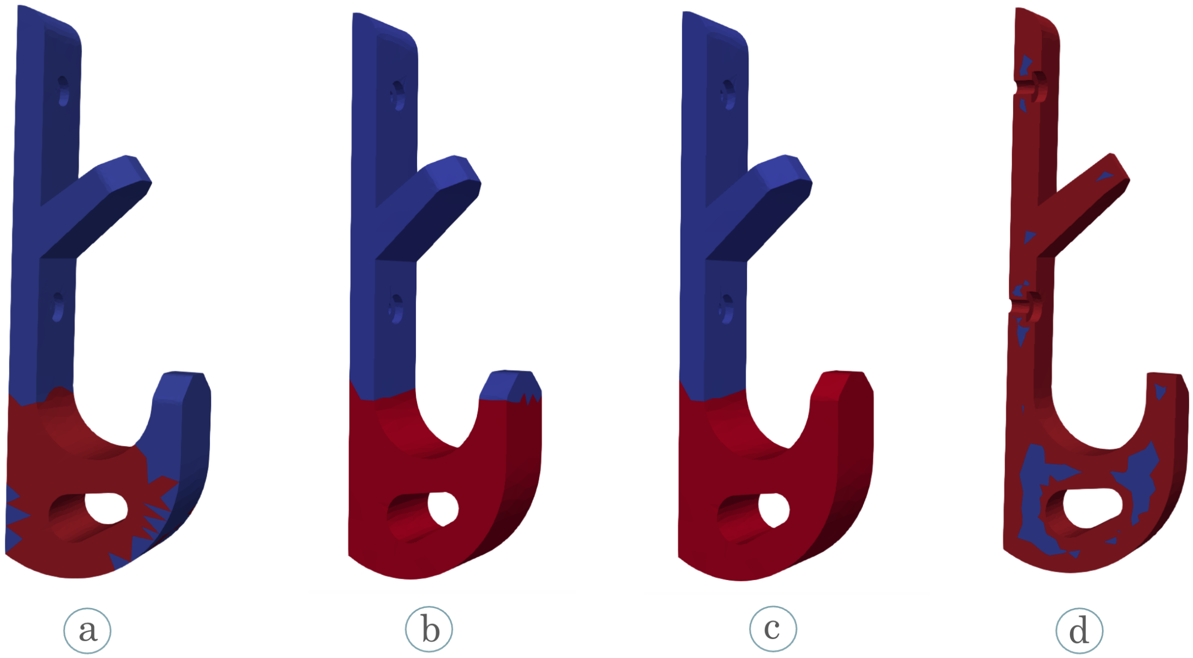


Figure 3.3: Segmentation comparison. **(a)** Layer-based segmentation partitions the model into horizontal bands, reinforcing full layers if stress exceeds a defined fraction. **(b)** Threshold-based segmentation directly maps stress magnitude to material, reinforcing only the highest-stress elements. **(c)** Optional perimeter shell adjustment applies a uniform outer layer of virgin PLA. **(d)** Island removal removes small disconnected fragments of high-strength material, such as the tip of the hook, for improved printability and strength.

Figure 3.3 illustrates the different segmentation strategies and post-processing heuristics applied to a hook model. These visualizations help demonstrate how user-configured heuristics shape the final material assignment.

3.4.1 Threshold-Based Segmentation

The first strategy directly maps high-stress regions to stronger material using a stress percentile threshold. This method assigns a binary material label to each tetrahedral element based on whether its stress exceeds a user-defined percentile of the global stress distribution. For instance, if the threshold is set to the 80th percentile, only the top 20% of stress values are marked for reinforcement.

This approach offers direct control over how much of the object will use virgin material and works particularly well for models where stress is highly localized. It is implemented by computing the percentile cutoff on the array of stress magnitudes and assigning material labels accordingly, as shown in Figure 3.3b.

3.4.2 Layer-Based Segmentation

To support objects with more distributed or complex stress patterns, we also implemented a layer-based segmentation method. This technique first divides the mesh into axis-aligned layers along its longest geometric dimension. Each tetrahedral element is assigned to a layer based on the mean coordinate of its vertices.

Then, for each layer, the algorithm checks whether a sufficient fraction of tetrahedra within that layer are above a stress percentile threshold. If the fraction of high-stress elements exceeds a configurable minimum, the entire layer is reinforced with virgin PLA.

This approach allows for better material cohesion across layers and produces segmentations that are easier to manufacture with fused deposition modeling (FDM) printers, where changing materials across layers is often more reliable than within layers. See Figure 3.3a for a visual comparison.

3.4.3 Post-Segmentation Adjustments

After the initial segmentation, the system applies optional post-processing steps based on user configuration. These include:

- **Perimeter Shell Adjustment:** A thin shell of stronger material is applied to the outer surface of the model to improve print durability and reduce delamination risks (Figure 3.3c).

- **Island Removal:** Small, disconnected clusters of a single material (especially fragile recycled material) are automatically detected and flipped to match their surrounding dominant material. This step ensures manufacturability and reduces failure risks in narrow or isolated regions—such as the high-stress island at the tip of the hook in Figure 3.3d.

3.4.4 User Interaction

To enable interactive exploration of segmentation strategies, the simulation and segmentation steps are decoupled in the codebase. Users first run the simulation once to generate stress data, then apply different segmentation settings via `user_config.yaml`. This design choice improves usability by enabling rapid experimentation with stress thresholds, layer counts, and material fraction constraints without the need to re-run expensive simulations.

3.4.5 Material Assignment

After segmentation, each tetrahedral element is labeled with one of two materials: a high-strength material (e.g., `Fresh_PLA`) or a low-strength material (e.g., `Recycled_PLA`). The user can specify which materials to assign to each region through configuration files. The final assignment is saved and used for mesh export and print preparation.

3.5 Segmentation Heuristics for Print Reliability

While stress-based material segmentation effectively reinforces high-stress regions of a 3D print, certain geometric or topological edge cases can still result in undesired print outcomes. To improve both the manufacturability and the visual consistency of resulting prints, the system incorporates two optional post-processing heuristics: **island removal** and **perimeter shell enforcement**. These heuristics are configurable and can be enabled independently depending on user preference and print context.

3.5.1 Island Removal Heuristics

Segmenting a 3D mesh into regions of strong and weak material can sometimes result in small, disconnected "islands" of high-strength material embedded within low-stress regions. These isolated tetrahedral clusters are often artifacts of local stress spikes, numerical noise, or thresholding artifacts. If left unaddressed, they introduce unnecessary material transitions, increase toolpath complexity during slicing, and risk weak mechanical bonding due to poor extrusion continuity.

To address this issue, we implement a suite of **island removal heuristics** that identify and reassign small, disconnected regions of material. The general strategy is to group tetrahedral elements into connected clusters using various neighborhood criteria or spatial similarity. Any region smaller than a user-defined threshold (based on a percentage of total mesh size) is flipped to match the dominant surrounding material, or to the opposite type by default.

In the following sections, we describe three methods we developed and tested:

- A flood-fill traversal that uses edge-sharing connectivity to detect islands.
- A graph-based method that uses connected components over a mesh adjacency graph.
- A K-Means clustering approach that leverages spatial proximity of tetrahedra to identify small regions.

Each of these methods has different trade-offs in terms of runtime, spatial awareness, and sensitivity to noise. By comparing their behavior across diverse geometries, we provide users with flexible and robust options to improve the quality of material segmentation.

Flood Fill-Based Island Removal

The flood fill method removes islands by traversing connected elements that share the same material type, grouping them into clusters, and flipping those clusters if they fall below a user-

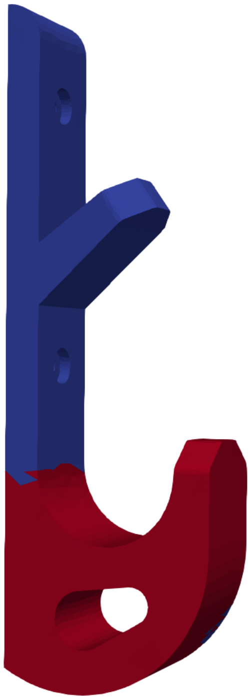


Figure 3.4: Flood-fill island removal result. Edge-connected clusters of virgin material (red) were evaluated for size and flipped when below threshold. This method effectively removes slender tendrils and small specks of high-strength material while preserving structural intent.

defined size threshold. Unlike purely face-connected approaches, this method considers any tetrahedra that share at least two vertices (i.e., an edge) as neighbors, improving sensitivity to thin or elongated regions that might otherwise be missed.

We begin by initializing all elements as unvisited and proceed with a breadth-first traversal. For each unvisited element, we launch a flood fill that recursively explores neighboring tetrahedra sharing the same material and at least two vertices. This process continues until all connected components of the same material type are labeled with a unique region ID.

After labeling all regions, the algorithm computes the size of each cluster. Any region whose size falls below a user-defined threshold (specified as a fraction of the total number of tetrahedra) is considered an island. These regions are flipped to the opposite material type (e.g., from virgin PLA to recycled PLA), reducing unnecessary filament transitions and enhancing structural cohesion.

This method proved robust across a variety of geometries, successfully eliminating slender tendrils or noise-generated outliers that may not have been detected by simpler face-sharing techniques (see Figure 3.4).

Graph-Based Island Removal

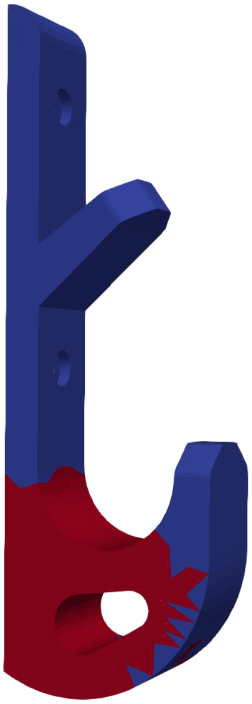


Figure 3.5: Graph-based island removal result. The red elements were originally disconnected clusters of high-strength material (e.g., virgin PLA) that were reassigned based on connectivity analysis. The method successfully eliminates noisy regions, especially near the edges of the hook.

The graph-based method identifies and removes small clusters of tetrahedral elements using a connected components algorithm. Each tetrahedron is modeled as a node in an undirected graph, and edges are created between nodes when their corresponding tetrahedra share a common face—defined as three or more shared vertices. This ensures that only face-connected elements of the same material are grouped together, which is a conservative but effective criterion for connectivity in 3D meshes.

Once the graph is constructed, connected components are extracted using the NetworkX library. Each connected component corresponds to a contiguous region of elements with the same material assignment. These components are then analyzed for size. Any region that falls below a user-defined minimum size threshold is considered an island.

Rather than analyzing local neighbors, the method simply flips the material assignment for all elements in the small region to the opposite material type. This approach helps eliminate noise-induced features and disconnected "specks" of high-strength material embedded within a low-strength field.

This method is conceptually simple and reliable for identifying clearly bounded regions. However, because it uses a face-sharing criterion, it may miss certain elongated or edge-connected features that are better detected by the flood fill method (see Figure 3.5).

Parameters:

- `island_max_percent`: the user-defined threshold that determines the minimum acceptable cluster size.
- `min_region_size`: computed as `max(num_tetra` × `island_max_percent, 5)`.

K-Means Clustering-Based Island Removal

The third method leverages `KMeans` clustering to identify and eliminate sparse, weak-material regions based on geometric proximity rather than topological connectivity. Unlike the graph-based and flood-fill approaches, which rely on vertex or face sharing, this method uses the spatial distribution of tetrahedral element centroids to cluster and filter disconnected or dispersed patches.

First, the centroid of each tetrahedral element is computed as the mean of its four vertex coordinates. K-means clustering is then performed specifically on the centroids of elements labeled as low-strength material (e.g., recycled PLA). This ensures that the algorithm focuses on small, potentially isolated weak regions embedded in a larger high-strength domain.

Each resulting cluster is evaluated by its size. If the number of elements in a cluster is below a user-defined threshold, it is deemed an island and its elements are flipped to the

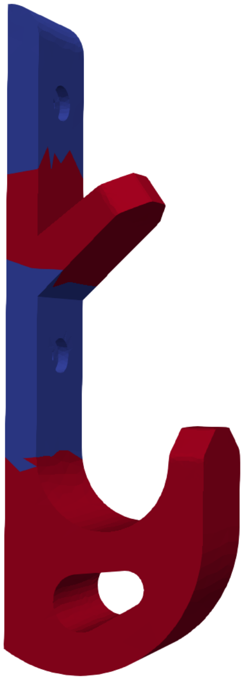


Figure 3.6: Segmentation result after applying K-means clustering-based island removal. Small, spatially isolated weak-material regions (blue) have been reassigned to strong material (red), producing smoother and more cohesive segmentation boundaries.

stronger material type. This method effectively removes dispersed speckles of weak material, even when such regions do not form contiguous groups by topological standards.

Figure 3.6 demonstrates the result of this clustering-based approach, showing how spatially separated low-strength islands have been reassigned for a more reliable and manufacturable print.

While this approach does not account for mesh connectivity, it is particularly effective in identifying clusters that are spatially isolated but not topologically well-separated—cases where previous methods may be overly conservative.

Parameters:

- `kmeans_num_clusters`: the number of centroid-based clusters to compute.
- `island_max_percent`: determines the minimum cluster size as a fraction of the total number of tetrahedra.

- `min_cluster_size`: the cutoff below which clusters are flipped to strong material.

3.5.2 Perimeter Shell Adjustment

Surface quality and visual uniformity are important for many 3D-printed parts. In practice, slight color or texture differences between recycled and virgin filaments can cause visible inconsistency if material switches occur near the object's exterior. To address this, the system allows users to apply a *uniform outer shell* of strong material, regardless of stress.

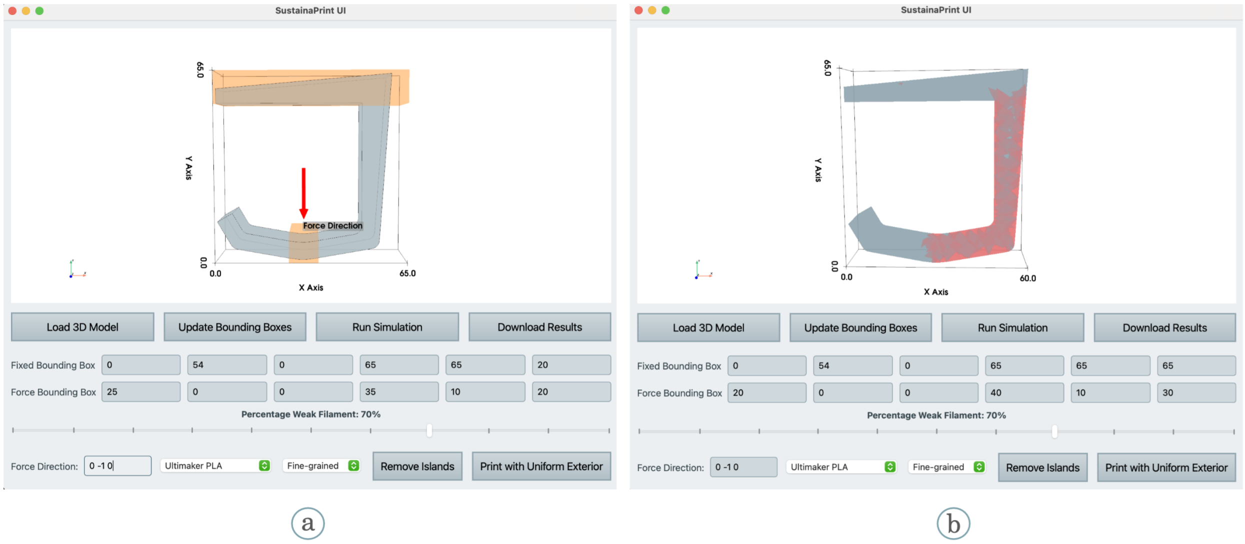
The perimeter adjustment heuristic identifies all tetrahedra that are adjacent to the model's surface. We define the surface as the set of triangular faces that are shared by only one tetrahedron. Any tetrahedral element that contains one or more nodes from these surface faces is marked as a surface element. These surface elements are then forcibly assigned to the high-strength material.

In addition to visual benefits, this strategy can improve inter-layer bonding near the object's perimeter and reduce the risk of cracks or delamination, particularly in dual-material FDM prints.

3.6 User Interface Integration

The user interface (UI), shown in Figure 3.7, enables users to explore different segmentation options and material assignment settings without needing to interact directly with the simulation codebase. The frontend was developed by collaborators on the project, while my contributions focused on designing a robust and modular backend that could seamlessly interface with the UI.

To support user-driven interaction, I organized the simulation pipeline around two YAML configuration files: `setup_config.yaml`, which defines simulation parameters such as bounding boxes, loading vectors, and number of steps, and `user_config.yaml`, which includes segmentation parameters like threshold levels, island removal options, and layer



ⓐ

Fixed Bounding Box: 0, 54, 0, 65, 65, 20
Force Bounding Box: 25, 0, 0, 35, 10, 20
Percentage Weak Filament: 70%
Force Direction: 0 -1 0
Ultimaker PLA
Fine-grained

ⓑ

Fixed Bounding Box: 0, 54, 0, 65, 65, 65
Force Bounding Box: 20, 0, 0, 40, 10, 30
Percentage Weak Filament: 70%
Force Direction: 0 -1 0
Ultimaker PLA
Fine-grained

Figure 3.7: (a) system UI, where users load a 3D model and specify its use case via bounding boxes and force direction. (b) After segmentation, users adjust the percentage of weak filament, select a segmentation mode, and download STL files for slicing and printing.

splitting criteria. This structure allows the frontend to update parameters independently of the simulation engine, making the system both extensible and user-friendly.

Each time a user updates the segmentation parameters—such as adjusting the percentage of weak filament, toggling island removal, or enabling a uniform exterior shell—the backend re-runs the segmentation logic and generates a new 3D preview. This real-time feedback loop lets users quickly iterate on their settings before finalizing their material assignments.

Once satisfied, the user can click the “Download Results” button to export the segmented mesh into STL and OBJ formats, one for each material region. These files are organized for direct compatibility with dual-extrusion 3D printers, where each region can be mapped to a specific filament.

By decoupling the simulation from the segmentation and preview logic, I enabled a workflow in which the full simulation only needs to be run once. Users can then experiment freely with segmentation methods—threshold-based or layer-based—and heuristics without incurring the cost of recomputing the stress field. This separation of concerns improves both efficiency and usability, supporting design iteration and parameter tuning on consumer hardware.

3.7 Interlocking Material Transitions

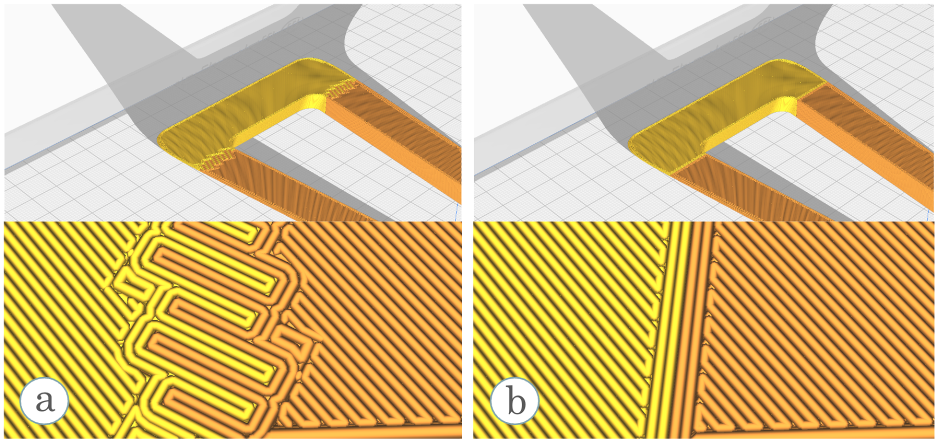


Figure 3.8: (a) Headphone stand sliced with interlocking material interfaces. (b) Without interlocks, showing a simple planar boundary prone to delamination.

Within the larger project this thesis is part of, inbuilt interlocking features are used as an important complementary technique used during slicing to improve adhesion between materials. Most 3D prints tend to fail at material boundaries, especially when recycled and virgin filaments have different extrusion characteristics. The *interlocking* feature creates comb-like patterns at these boundaries, increasing contact area and helping to prevent delamination.

As shown in Figure 3.8, this approach significantly enhances interface strength with minimal user effort, and is critical for prints where high-stress regions lie near material transitions. Because it is applied at the slicing stage, it integrates seamlessly with the output of our segmentation pipeline.

Chapter 4

Applications

This chapter highlights several example applications that demonstrate how our pipeline can be used to reinforce structurally critical regions while maximizing the use of recycled filament. As part of the larger research project, the collaborators printed several applications. These real-world examples showcase the versatility of the system across different geometries and loading conditions. Each object was analyzed using the pipeline described in Chapter 3, and materials were segmented based on predicted stress distributions. All examples retain up to 80% recycled filament composition, balancing sustainability with functional durability.

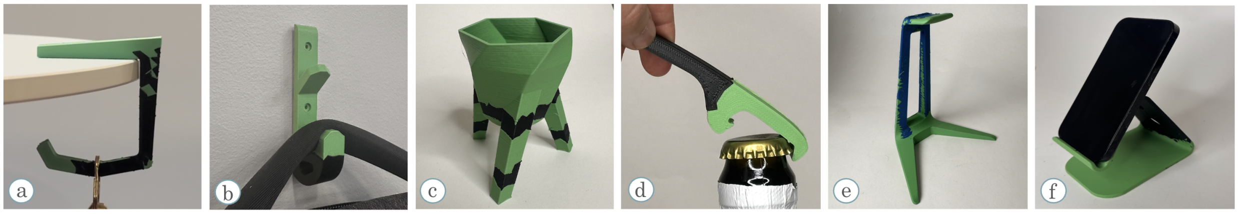
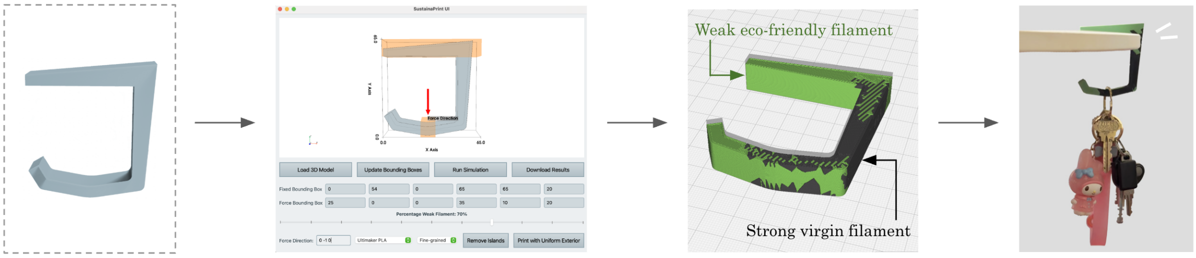


Figure 4.1: Applications. Can be applied to a wide range of everyday items, including reinforced hooks, plant pots, bottle openers, and headphone stands.



ⓐ **Input Model** ⓑ **SustainaPrint UI** ⓒ **Slice and Print** ⓓ **Use**

Figure 4.2: Workflow illustrated on a table hook. (a) The user begins with a 3D model of the object. (b) In the UI, the user simulates the stress distribution and specifies the desired balance between eco-friendly and virgin filament. (c) The model is automatically segmented and exported for multi-material slicing, assigning virgin filament to high-stress regions and eco-friendly filament to the remainder. (d) The final printed object supports real-world loads while minimizing the use of virgin plastic.

4.1 Hooks and Hangers

Hooks are popular 3D-printed items that often experience concentrated loading at the mounting point and hook tip. With this system, users can specify the fixed and loaded bounding boxes to simulate wall-mounted or table-mounted hooks. The simulation identifies high-stress regions at the base and curve of the hook, which are then reinforced with virgin PLA. This ensures the hook can safely carry weight without overusing strong filament elsewhere.

4.2 Standing Planters

Standing objects with legs—such as plant pots—are prone to stress concentrations where the legs meet the base. By simulating the weight of soil or water distributed across the top and solving for internal stresses, we assign strong filament to the high-load regions near the base while allowing the pot walls and upper legs to be printed in recycled PLA. This results in a lightweight yet sturdy structure with minimal virgin material usage.

4.3 Bottle Openers

Bottle openers require high rigidity in specific regions due to torque applied at the fulcrum and prying edge. Using a simple simulation setup—fixing one end and applying a force at the opposite tip—the pipeline highlights zones of maximal bending stress. These zones are reinforced with strong filament while the remainder of the object is printed with eco-friendly material. This strategy maintains functionality while reducing environmental impact.

4.4 Stands and Supports

We tested the pipeline on common accessories like phone and headphone stands, which are required to support vertical loads over time. Simulation allows users to target regions such as cantilevered hooks and central spines, which often exhibit the highest stress. These regions are assigned strong material, while the base and low-stress areas use recycled PLA. The result is a durable yet sustainable alternative to fully virgin prints.

Chapter 5

Evaluation

This chapter presents the evaluation of the segmentation and simulation pipeline developed in this thesis. First, I assess the performance of the stress-based segmentation methods, including runtime analysis, segmentation effectiveness, island removal techniques, and validation against commercial finite element software (Abaqus). These experiments represent my individual contributions.

Second, I summarize mechanical testing results from the broader team, which demonstrate the structural benefits of stress-aware material assignment in real-world objects. The experimental tests presented in this section were conducted by other members of the broader project team. While these results are important for validating the overall feasibility of stress-based segmentation with recycled materials, they do not represent my direct contributions.

5.1 Segmentation Evaluation

This section evaluates the performance of the segmentation and simulation pipeline developed in this thesis. I focus on four primary aspects:

- **Runtime Analysis:** Measuring the computational cost of key stages including meshing, simulation, stress analysis, segmentation, and island removal.

- **Stress Segmentation Effectiveness:** Evaluating how well the stress-based segmentation methods (threshold-based and layer-based) separate high- and low-stress regions.

- **Island Removal Performance:** Comparing three different island removal heuristics (flood fill, graph-based, and k-means clustering) based on runtime and segmentation quality.

- **Simulation Validation:** Comparing stress, strain, and displacement fields produced by the SfePy pipeline against Abaqus simulations to verify mechanical fidelity.

Each of these evaluations highlights the trade-offs and design choices involved in building a practical, accessible tool for sustainable fabrication.

5.1.1 Runtime Analysis

To evaluate the efficiency of the segmentation pipeline, we recorded runtimes for each major stage of the workflow across a diverse set of test models. These timings include meshing, simulation, and segmentation, and are plotted against the number of tetrahedral elements in Figure 5.1.

As expected, runtime increases roughly linearly with the number of tetrahedra, with simulation (solving) dominating the overall computational cost. Meshing and segmentation stages are comparatively lightweight, though segmentation time becomes more variable for larger models.

The segmentation times shown in Figure 5.1 reflect the use of the graph-based island removal method, which prioritizes conservative island detection by requiring face connectivity. While effective, this approach introduces modest overhead, especially for dense meshes.

To improve performance in time-sensitive scenarios, we evaluated a centroid-based alternative using KMeans clustering (see Section 3.5.1). This method operates directly on tetrahedral

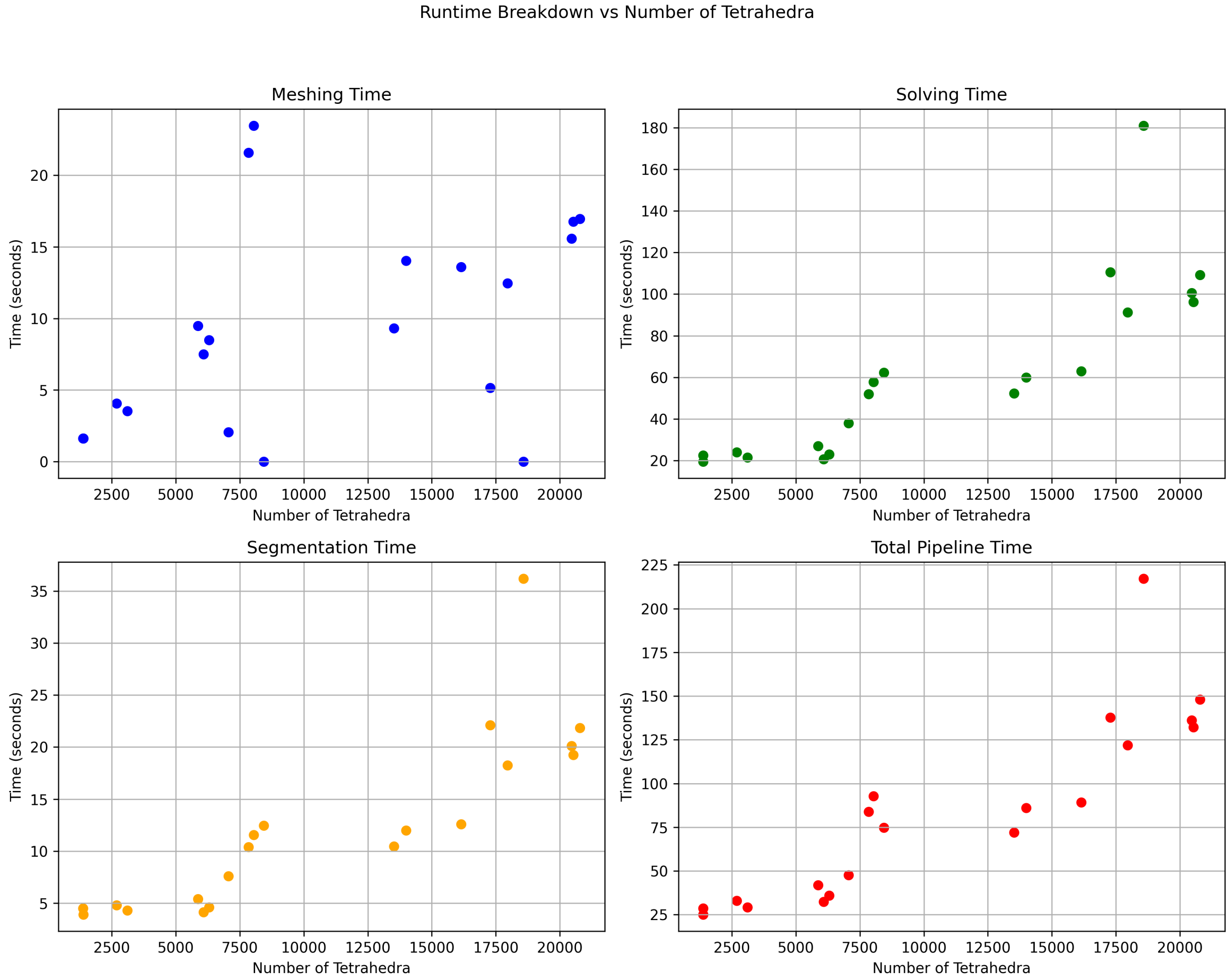


Figure 5.1: Runtime breakdown of the full segmentation pipeline across models of varying mesh size. Each point corresponds to a complete run of the pipeline on a single model, showing meshing, solving, segmentation, and total time versus number of tetrahedral elements.

centroids and does not require full topological traversal. In practice, it significantly reduced segmentation time for large meshes.

Overall, the pipeline exhibits acceptable scalability for interactive use. Additionally, the solving stage only must be performed once. This means that the user must only wait for the solver, then the user can adjust the segmentation thresholds (which only takes seconds).

5.1.2 Island Removal Evaluation

To assess the performance of our island removal heuristics, we implemented and evaluated three methods: flood-fill, graph-based connected components, and K-means centroid clustering. Our goal was to analyze the computational efficiency of each approach and to identify tradeoffs in runtime and segmentation quality.

Runtime Comparison

To assess the computational efficiency of different island removal strategies, we benchmarked the runtime of all three methods (graph-based, flood fill, and K-means clustering) on a single tetrahedral mesh at increasing sizes. For each method, we randomly sampled subsets of the mesh ranging from 10% to 100% of the tetrahedra and recorded the total execution time required to identify and reassign small material islands.

Figure 5.2 shows the results. Both the graph-based (NetworkX) and flood-fill methods exhibit roughly quadratic scaling with the number of tetrahedra, reflecting the cost of face or edge connectivity checks and recursive traversal. The graph-based approach is the slowest, due to the overhead of building a full adjacency graph.

By contrast, K-means clustering remains nearly constant in runtime. This is expected, as the clustering is only performed on weak-material elements and scales with the number of centroids rather than the full mesh size. This makes K-means a practical choice for large meshes where topological traversal is costly.

Overall, the runtime trade-offs mirror the conceptual strengths of each approach: while graph-based and flood-fill methods operate on mesh connectivity and may yield more accurate topological cleanup, K-means provides a much faster geometric approximation—suitable when speed is critical.

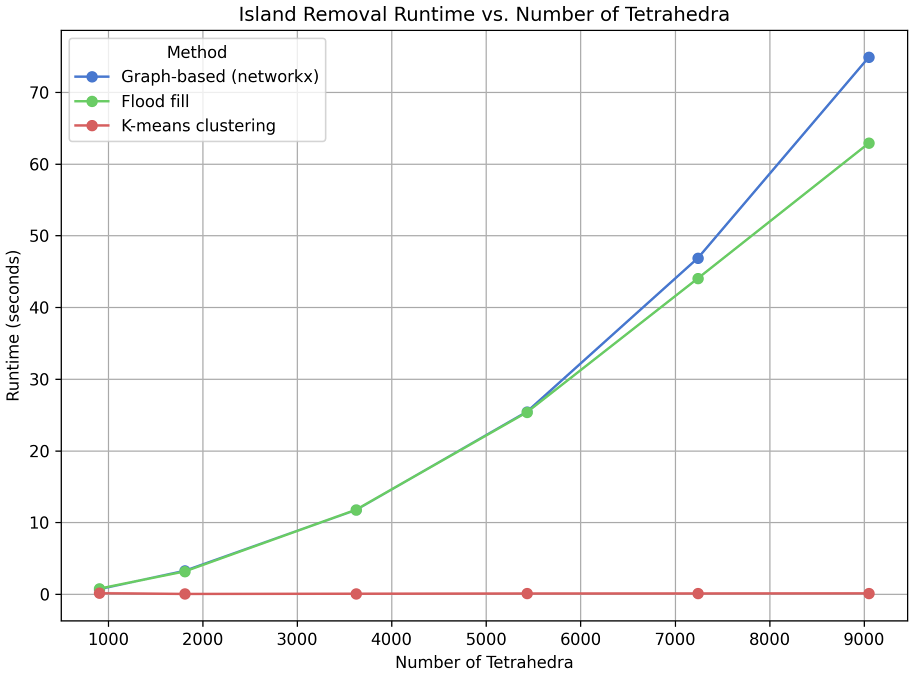


Figure 5.2: Runtime of island removal methods as a function of tetrahedral mesh size. The graph-based and flood fill methods scale quadratically with mesh size, while K-means clustering remains effectively constant.

Island Removal Effectiveness

To evaluate the quality of island removal, we measured the total number of connected material regions (including both strong and weak filament assignments) that remain after applying each method across a range of island size thresholds. Tetrahedra were considered part of the same group if they shared a face and were assigned the same material. Fewer groups generally indicate more cohesive material zones, which can lead to smoother 3D printing and stronger bonding across adjacent regions.

Figure 5.3 shows the number of connected groups remaining after island removal using the Graph-Based (NetworkX), Flood Fill, and K-Means Clustering methods. As expected, increasing the island size threshold results in more aggressive merging of small disconnected clusters, reducing the total number of groups. This is up to a certain point (Graph-Based and Flood Fill), where we eventually do not remove any islands because none of them are large enough to be considered islands.

The K-Means method consistently produced the fewest material groups across most thresholds. This is due to its global clustering approach, which segments weak-material regions based on centroid proximity rather than mesh connectivity. While this can be computationally efficient (as discussed in the runtime analysis), it may produce regions that are spatially coherent but not necessarily topologically connected.

The Graph-Based and Flood Fill methods, which operate directly on mesh topology, performed similarly at higher thresholds. However, they show more variability at lower thresholds, where small disconnected regions persist. These methods maintain stricter geometric boundaries, which can be useful when mechanical connectivity is critical but may result in more fragmented prints when thresholds are too conservative.

Importantly, the system supports real-time preview of segmentation results for different threshold values. This allows users to iterate and inspect material boundaries before committing to final STL or OBJ export, making it practical to balance between cohesion, print time, and performance without rerunning the full simulation pipeline.

5.1.3 SfePy and Abaqus Comparison

To verify the accuracy and reliability of our simulation pipeline, we qualitatively compared the field outputs of our custom SfePy-based simulation framework against those produced by Abaqus—an industry-standard finite element analysis (FEA) tool.

Rather than focusing on precise numerical agreement (which may vary due to differences in meshing, solvers, or boundary handling), we examined the overall stress, strain, and displacement distributions for the same test geometry and loading conditions. These visual comparisons provide confidence that our SfePy setup reproduces the core mechanical behavior seen in commercial tools.

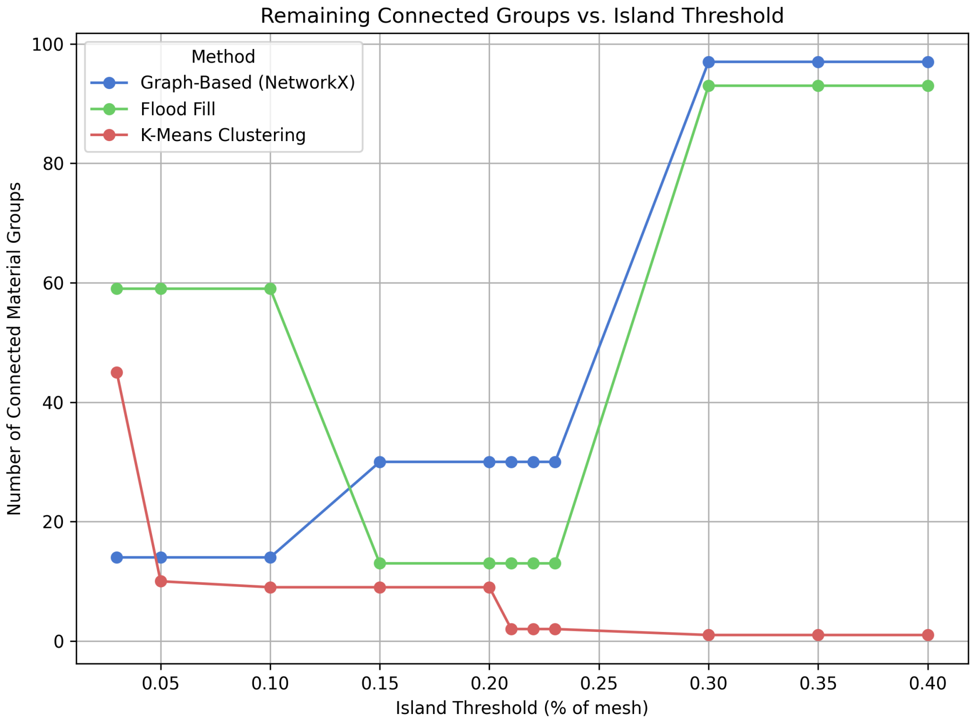


Figure 5.3: Remaining number of connected material regions after island removal, as a function of island size threshold. Lower values indicate more cohesive material assignments.

Visualization Approach

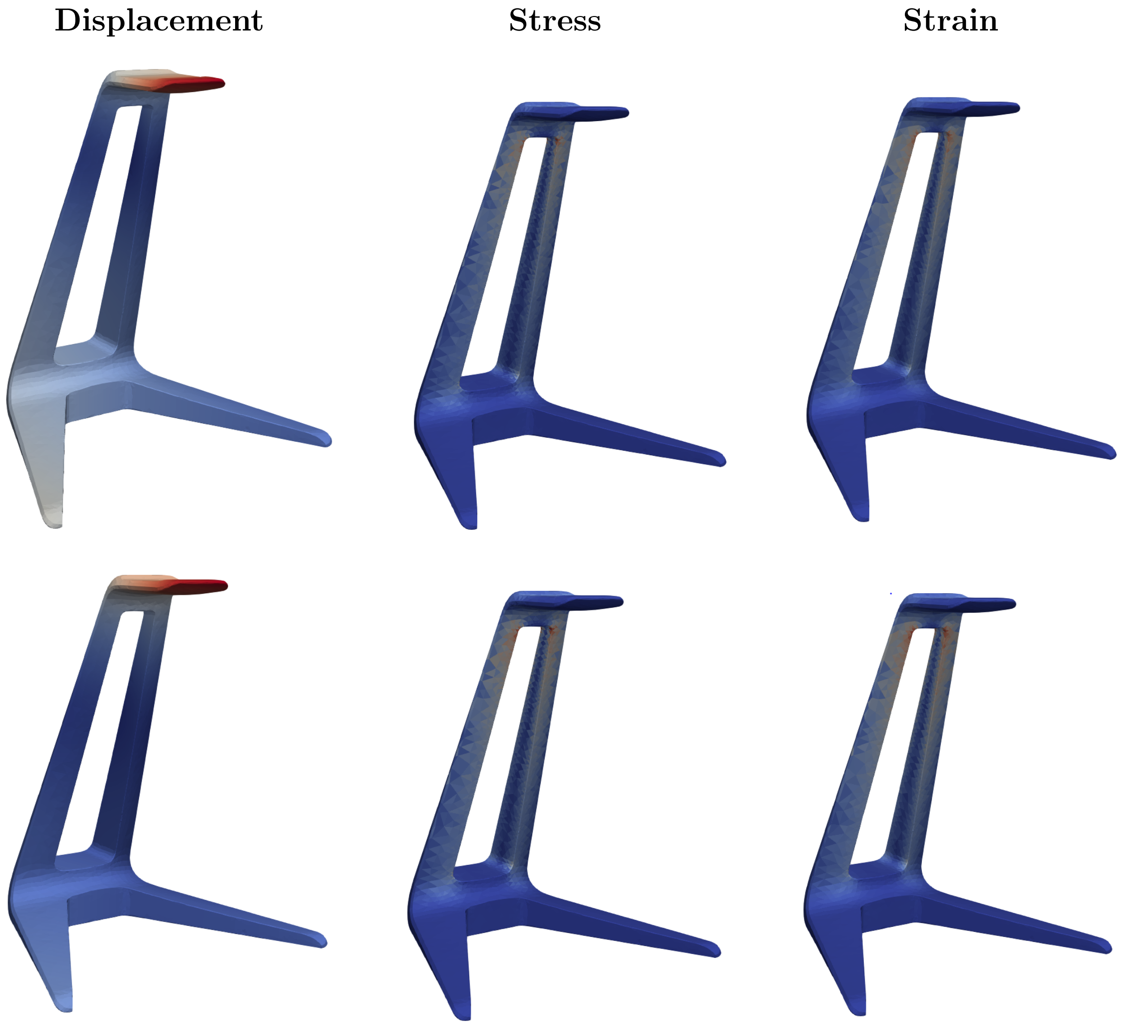
For both tools, we extracted and visualized the following quantities:

- Von Mises stress distribution
- Principal strain field (maximum component)
- Total displacement magnitude

Each field was rendered as a heatmap over the tetrahedral mesh, providing a spatial overview of internal loading responses. Figure 5.4 presents these comparisons side by side.

Observations

In all three fields, the patterns produced by SfePy align well with those from Abaqus. High-stress and high-strain regions consistently appeared in the same geometric areas, and



Displacement **Stress** **Strain**

Figure 5.4: Comparison of simulation results between SfePy (top row) and Abaqus (bottom row) for the same geometry and loading condition. Each column shows a different physical field: total displacement (left), von Mises stress (center), and principal strain (right). Visual alignment across tools demonstrates the fidelity of the SfePy simulation pipeline.

displacement gradients followed matching trends. These visual correlations suggest that the simulation logic implemented in SfePy—including material modeling, boundary conditions, and loading—closely approximates the behavior predicted by a commercial solver.

This result supports our decision to use SfePy as the simulation backend for our sustainable material assignment pipeline, enabling flexible, open-source experimentation without sacrificing fidelity.

5.2 Experimental Testing on Physical Prints (Prior Work)

The experimental tests presented in this section were conducted by other members of the broader project team. While these results are important for validating the overall feasibility of stress-based segmentation with recycled materials, they do not represent my direct contributions.

5.2.1 Primitive Geometries

The team tested five primitive shapes—hook, ring, beam, cylinder, and dome (Figure 5.5)—chosen to reflect canonical structural load cases such as tension, compression, and bending.

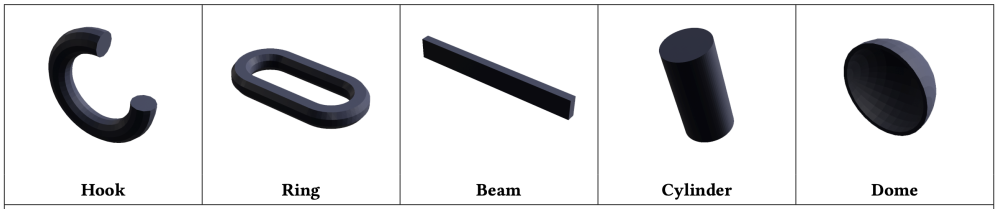
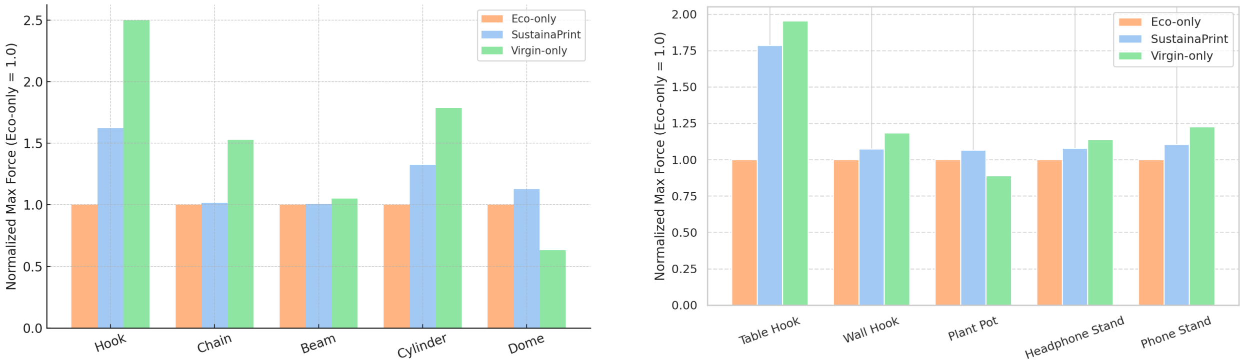


Figure 5.5: Primitive geometries evaluated under mechanical loading: hook, ring, beam, cylinder, and dome.

Objects were printed using three material configurations: fully eco-friendly PLA, a hybrid stress-informed assignment, and fully virgin PLA. Across all shapes, hybrid material assignment significantly improved load-bearing capacity compared to eco-only prints, achieving performance roughly halfway between eco-only and fully virgin prints (Figure 5.6).

5.2.2 Functional Geometries

Similar trends were observed for functional objects such as table hooks, headphone stands, and phone stands. Stress-informed material assignment resulted in notable improvements in strength and toughness, while maintaining a high proportion of recycled filament. Some variation in stiffness was observed depending on the geometry.



(a) Primitive objects.

(b) Functional objects.

Figure 5.6: Comparison of maximum force sustained across material configurations. Hybrid prints consistently outperform eco-only prints while using limited virgin material.

5.2.3 Case Study: Tripod Plant Pot

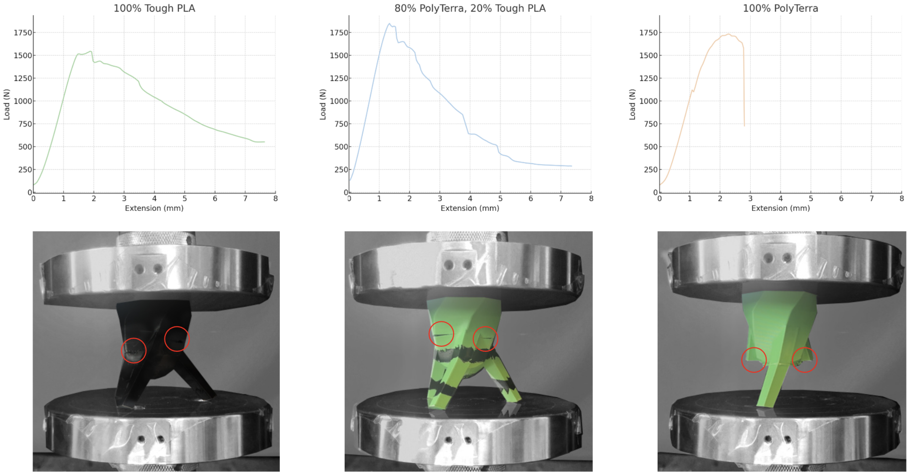


Figure 5.7: Failure comparison for the tripod plant pot. Left: 100% eco-friendly print showing brittle fracture. Middle: Hybrid assignment redistributing stress and delaying failure. Right: Virgin PLA print.

In the tripod plant pot (Figure 5.7), fully eco-friendly prints fractured at only 2.8 mm of compression. Hybrid material assignment successfully reinforced stress-critical regions, delaying buckling and producing a smoother, more resilient failure mode. Performance closely matched that of the fully virgin prints, demonstrating the value of stress-informed material assignment for structural durability.

Summary

Although this experimental evaluation was not part of my direct contributions, it provides supporting evidence that stress-driven hybrid printing can significantly improve mechanical performance while reducing reliance on virgin material.

Chapter 6

Conclusion

This thesis presents a simulation-driven pipeline for assigning recycled and virgin PLA to different regions of a 3D print based on predicted stress distributions. By using finite element simulation to inform segmentation, the system helps makers retain structural performance while maximizing the use of sustainable materials.

The pipeline integrates open-source tools for meshing, simulation, segmentation, and post-processing, and supports user-configurable options for load direction, material models, and segmentation strategy. Optional enhancements such as island removal and shelling improve both print quality and durability.

While I did not contribute to the mechanical testing presented in the broader team evaluation, my work focuses on enabling robust material assignment through stress simulation, and on evaluating trade-offs across different segmentation strategies. In particular, I developed and compared several island removal techniques—graph-based, flood fill, and centroid-based clustering—and evaluated their effectiveness and runtime performance.

Key contributions include:

- A modular simulation pipeline for stress-aware material segmentation.
- Support for threshold-based, layer-based, and post-processing segmentation strategies.

- An evaluation of multiple island removal techniques, including performance and visual outcomes.
- A comparison of simulation outputs between an open-source solver (SfePy) and commercial software (Abaqus), confirming fidelity.

Limitations. The current framework supports a limited set of material models and requires manual setup of loading conditions and region selection. Results may vary depending on mesh resolution, solver accuracy, and filament variability.

Future Work. Opportunities include automatic detection of loading regions, integration with slicing software, and support for broader filament types and extrusion strategies. A user study could further validate ease of use and impact on print outcomes.

By lowering the barrier to incorporating recycled filament in functional designs, this work helps advance sustainable personal fabrication. The open-source release aims to support further research, prototyping, and educational use.

Appendix A

Code Repository

Please contact Cole Paulin for code.

References

[1] C. DiSalvo, P. Sengers, and H. Brynjarsdóttir. “Mapping the Landscape of Sustainable HCI”. In: *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. 2010, pp. 1975–1984.

[2] E. Barba, E. Walker, A. Y. Wang, J. Turns, D. Rosner, and J. E. Froehlich. “Make Making Sustainable: Exploring Sustainability Practices, Challenges, and Opportunities in Making Activities”. In: *Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems*. Association for Computing Machinery, 2022, pp. 1–16. DOI: 10.1145/3491102.3502073.

[3] F. Encinas, M. Pimenta, F. de Souza, M. Wanderley, E. Reategui, and L. Nedel. “Introducing the Sustainable Prototyping Life Cycle for Digital Fabrication to Designers”. In: *Proceedings of the 2020 ACM Designing Interactive Systems Conference (DIS ’20)*. 2020, pp. 1079–1093. DOI: 10.1145/3357236.3395510.

[4] M. R. Hasan, I. J. Davies, A. Pramanik, M. John, and W. K. Biswas. “Potential of Recycled PLA in 3D Printing: A Review”. In: *Sustainable Manufacturing and Service Economics* (2024), p. 100020.

[5] S. Budin, N. C. Maideen, M. H. Koay, D. Ibrahim, and H. Yusoff. “A Comparison Study on Mechanical Properties of Virgin and Recycled Polylactic Acid (PLA)”. In: *Journal of Physics: Conference Series*. Vol. 1349. IOP Publishing, 2019, p. 012002.

[6] J. Zhu, L. Winagle, and H.-L. Kao. “EcoThreads: Prototyping Biodegradable E-Textiles through Thread-Based Fabrication”. In: *Proceedings of the CHI Conference on Human Factors in Computing Systems*. 2024, pp. 1–17.

[7] L. W. Wall, A. Jacobson, D. Vogel, and O. Schneider. “Scrappy: Using Scrap Material as Infill to Make Fabrication More Sustainable”. In: *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems*. 2021, pp. 1–12.

[8] X. A. Chen, S. Coros, J. Mankoff, and S. E. Hudson. “Encore: 3D Printed Augmentation of Everyday Objects with Printed-over, Affixed and Interlocked Attachments”. In: *Proceedings of the 28th Annual ACM Symposium on User Interface Software Technology*. 2015, pp. 73–82.

[9] F. R. Beltrán, C. Infante, M. U. de la Orden, and J. M. Urreaga. “Mechanical Recycling of Poly (Lactic Acid): Evaluation of a Chain Extender and a Peroxide as Additives for Upgrading the Recycled Plastic”. In: *Journal of Cleaner Production* 219 (2019), pp. 46–56.

[10] R. Song and C. Telenko. “Material waste of commercial FDM printers under realistic conditions”. In: *Proceedings of the 27th Annual International Solid Freeform Fabrication Symposium*. 2016, pp. 1217–1229.

[11] E. Gkartzou, E. P. Koumoulos, and C. A. Charitidis. “Production and 3D Printing Processing of Bio-Based Thermoplastic Filament”. In: *Manufacturing Review* 4 (2017), p. 1.

[12] M. L. Rivera, S. S. Bae, and S. E. Hudson. “Designing a Sustainable Material for 3D Printing with Spent Coffee Grounds”. In: *Proceedings of the 2023 ACM Designing Interactive Systems Conference*. 2023, pp. 294–311.

[13] R. Cimrman et al. *SfePy - Simple Finite Elements in Python*. Version 2019.4. 2019. URL: http://sfepy.org.

[14] D. Hibbitt, P. Sorensen, and B. Karlsson. *ABAQUS/Standard: User's Manual*. Providence, RI, USA: Hibbitt, Karlsson & Sorensen, 2001.

[15] Y. Hu, Q. Zhou, X. Gao, A. Jacobson, D. Zorin, and D. Panozzo. "Tetrahedral Meshing in the Wild". In: *ACM Transactions on Graphics (TOG)* 37.4 (2018), pp. 1–14.

[16] J. Ahrens, B. Geveci, and C. Law. "ParaView: An End-User Tool for Large-Data Visualization". In: *The Visualization Handbook*. Elsevier, 2005, pp. 717–731.

[17] O. C. Zienkiewicz, R. L. Taylor, and J. Z. Zhu. *The Finite Element Method for Solid and Structural Mechanics*. 6th. Elsevier Butterworth-Heinemann, 2005. ISBN: 9780750663212.

[18] F. Encinas, M. Pimenta, F. de Souza, M. Wanderley, E. Reategui, and L. Nedel. "Introducing the Sustainable Prototyping Life Cycle for Digital Fabrication to Designers". In: *Proceedings of the 2020 ACM Designing Interactive Systems Conference (DIS '20)*. New York, NY, USA: Association for Computing Machinery, 2020, pp. 1079–1093. DOI: 10.1145/3357236.3395510. URL: https://doi.org/10.1145/3357236.3395510.

[19] J. Lu, B. Desta, K. D. Wu, R. Nith, J. E. Passananti, and P. Lopes. "ecoEDA: Recycling E-waste During Electronics Design". In: *Proceedings of the 36th Annual ACM Symposium on User Interface Software and Technology (UIST '23)*. New York, NY, USA: Association for Computing Machinery, 2023, pp. 1–14. DOI: 10.1145/3586183.3606745. URL: https://doi.org/10.1145/3586183.3606745.

[20] R. Zhang, S. Mueller, G. L. Bernstein, A. Schulz, and M. Leake. "Wastebanned: Supporting zero waste fashion design through linked edits". In: *Proceedings of the 37th Annual ACM Symposium on User Interface Software and Technology*. 2024, pp. 1–13.

[21] J. Froehlich, T. Dillahunt, P. Klasnja, J. Mankoff, S. Consolvo, B. Harrison, and J. A. Landay. "UbiGreen: investigating a mobile tool for tracking and supporting green transportation habits". In: *Proceedings of the sigchi conference on human factors in computing systems*. 2009, pp. 1043–1052.

[22] E. Barba, E. Walker, A. Y. Wang, J. Turns, D. Rosner, and J. E. Froehlich. "Make Making Sustainable: Exploring Sustainability Practices, Challenges, and Opportunities in Making Activities". In: *Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems*. New York, NY, USA: Association for Computing Machinery, 2022, pp. 1–16. DOI: 10.1145/3491102.3502073. URL: https://doi.org/10.1145/3491102.3502073.

[23] S. Mueller, S. Im, S. Gurevich, A. Teibrich, L. Pfisterer, F. Guimbretière, and P. Baudisch. "WirePrint: 3D printed previews for fast prototyping". In: *Proceedings of the 27th Annual ACM Symposium on User interface Software and Technology*. 2014, pp. 273–280.

[24] A. Jones, T. Cameron, B. Eichholz, M. Eichers, T. Kray, and J. Straub. "Reducing space sensing and other mission cost with 3D printing infill optimization". In: *Sensors and Systems for Space Applications XII*. Vol. 11017. SPIE. 2019, pp. 69–74.

[25] K. Zhu, A. Dancu, and S. Zhao. "FusePrint: A DIY 2.5D printing technique for good-fit fabrication with daily objects". In: *Proceedings of the 2017 CHI Conference Extended Abstracts on Human Factors in Computing Systems*. 2017, pp. 413–416.

[26] R. Kovacs, A. Seufert, L. Wall, H.-T. Chen, F. Meinel, W. Müller, S. You, M. Brehm, J. Striebel, Y. Kommana, et al. "Trussfab: Fabricating sturdy large-scale structures on desktop 3D printers". In: *Proceedings of the 2017 CHI conference on human factors in computing systems*. 2017, pp. 2606–2616.

[27] M. L. Rivera, S. S. Bae, and S. E. Hudson. "Designing a sustainable material for 3D printing with spent coffee grounds". In: *Proceedings of the 2023 ACM Designing Interactive Systems Conference*. 2023, pp. 294–311.

[28] J. Zhu, L. Winagle, and H.-L. Kao. "EcoThreads: Prototyping biodegradable e-textiles through thread-based fabrication". In: *Proceedings of the CHI Conference on Human Factors in Computing Systems*. 2024, pp. 1–17.

[29] P. Gough, P. B. Perera, M. A. Kertesz, and A. Withana. “Design, Mould, Grow!: A fabrication pipeline for growing 3D designs using myco-materials”. In: *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems*. 2023, pp. 1–15.

[30] E. Gkartzou, E. P. Koumoulos, and C. A. Charitidis. “Production and 3D printing processing of bio-based thermoplastic filament”. In: *Manufacturing Review* 4 (2017), p. 1.

[31] D. Saakes, T. Cambazard, J. Mitani, and T. Igarashi. “PacCAM: Material Capture and Interactive 2D Packing for Efficient Material Usage on CNC Cutting Machines”. In: *Proceedings of the 26th Annual ACM Symposium on User Interface Software and Technology (UIST ’13)*. New York, NY, USA: Association for Computing Machinery, 2013, pp. 441–446. DOI: 10.1145/2501988.2501990. URL: https://doi.org/10.1145/2501988.2501990.

[32] A. H. Bermano, T. Funkhouser, and S. Rusinkiewicz. “State of the art in methods and representations for fabrication-aware design”. In: *Computer Graphics Forum* 36.2 (2017), pp. 509–535. DOI: 10.1111/cgf.13146.

[33] F. Faruqi, A. Katary, T. Hasic, A. Abdel-Rahman, N. Rahman, L. Tejedor, M. Leake, M. Hofmann, and S. Mueller. “Style2Fab: Functionality-Aware Segmentation for Fabricating Personalized 3D Models with Generative AI”. In: *Proceedings of the 36th Annual ACM Symposium on User Interface Software and Technology (UIST)*. 2023. DOI: 10.1145/3586183.3606723. URL: https://dl.acm.org/doi/10.1145/3586183.3606723.

[34] F. Faruqi, A. Katary, T. Hasic, A. Abdel-Rahman, N. Rahman, L. Tejedor, M. Leake, M. Hofmann, and S. Mueller. “Style2Fab: Functionality-Aware Segmentation for Fabricating Personalized 3D Models with Generative AI”. In: *Proceedings of the 36th Annual ACM Symposium on User Interface Software and Technology (UIST)*. ACM, 2023. DOI: 10.1145/3586183.3606723.

[35] T. G. Gutowski, S. Suh, E. M. Sekulic, and J. C. H. King. “Environmental Analysis of Manufacturing Processes”. In: *Environmental Science Technology* 45.10 (2011), pp. 4222–4229.

[36] J. M. F. Mendoza, A. Gallego-Schmid, X. Bengoa, and A. Azapagic. “Environmental Life Cycle Assessment of 3D Printing for the Manufacture of Consumer Products”. In: *Journal of Cleaner Production* 212 (2019), pp. 55–68.

[37] A. A. Escalante and D. A. Roberson. “Sustainability Assessment of Fused Filament Fabrication for Recycled and Virgin PLA Filaments”. In: *Sustainable Materials and Technologies* 29 (2021), e00301.

[38] K. Vidimče, S. Misailovic, F. Durand, and K. Z. Gajos. “Fab Forms: Customizable Objects for Fabrication with Validity and Geometry Constraints”. In: *Proceedings of the 26th Annual ACM Symposium on User Interface Software and Technology (UIST)*. 2013, pp. 13–22.

[39] C. Bader, M. Kolb, M. Dikovsky, and J. Oxman. “Making Data Matter: Voxel Printing for Digital Design and Fabrication”. In: *Proceedings of the 2016 ACM SIGGRAPH Conference*. 2016, pp. 1–12.

[40] A. Zoran and J. Paradiso. “Hyperform: Enabling Complexity through Incremental Fabrication”. In: *Proceedings of the 2013 ACM SIGCHI Conference on Human Factors in Computing Systems*. 2013, pp. 379–388.

[41] H.-T. Yang, C. Xiang, G. Laput, and C. Harrison. “MakerLens: Embedded Cameras for Visualizing Printing Process on Curved Surfaces”. In: *Proceedings of the 32nd Annual ACM Symposium on User Interface Software and Technology (UIST)*. 2019, pp. 461–473.

[42] A. Mclean, B. Bickel, and S. Mueller. “Printy: Interactive Printing on 3D Surfaces”. In: *Proceedings of the 2018 ACM Symposium on User Interface Software and Technology (UIST)*. 2018, pp. 679–690.