Design for Additive Manufacturing: Proposing a Design Methodology for Streamlining L-PBF Additive Manufacturing and Automation Potential

Kais Alhasan, Akshay Awaradi, Linus Frostell, Erik Gottfridsson,
Sushil Krishna Vadakkekara Puthan Veedu,
Erik Lindkvist

Project Course Advanced – Design Engineering and Product Development
Linköping University, January 2026

Abstract

Additive Manufacturing (AM), and its subcategory, Laser Powder Bed Fusion (L-PBF), enables the production of highly complex and lightweight components, but often requires time-consuming and fragmented design workflows involving multiple software tools. This project proposes a structured Design for Additive Manufacturing (DfAM) methodology, combined with partial automation, to reduce manual effort and improve process efficiency, streamlining the DfAM process.

The project is based on a systematic literature study, a comparative manual evaluation of selected CAD, simulation, and optimization software, and the definition of a representative test component. Based on these activities, a design methodology and a conceptual Python-based automation framework are proposed, aiming to connect and streamline key stages of the DfAM workflow such as design, simulation, and evaluation.

The result highlights significant differences in software capabilities, usability, and automation potential, with certain tools being more suitable for integration into automated workflows than others. While complete automation was not achieved within the scope of this project, the findings demonstrate clear opportunities for streamlining iterative DfAM processes and provide a foundation for future development of integrated and scalable automation frameworks.

Contents

1 Introduction

1.1 Background

AM, or Additive Manufacturing, is the process of manufacturing components through material being added layer by layer. Additive manufacturing with metal materials is most often done using a powder bed and a high-energy beam, fusing metal particles together, like in an L-PBF. AM is a manufacturing process that makes flexibility, customization, and personalization more available compared to other manufacturing methods, but comes with its own set of challenges. For instance, the process of making an additively manufactured part is often long and involves several different software programs, such as CAD, optimization, 3D print preparation, and verification software. The process of Designing for Additive Manufacturing (DfAM), which involves optimizing manufacturing, testing, assembly, service, repair, and cost without compromising safety, time to market, functionality, or styling includes numerous time-consuming steps. This project therefore aims to reduce the time needed and streamline the process of DfAM by proposing an optimized design methodology and automating parts of the process.

1.2 Project goals

The goal of this project is to reduce the time and steps required for and to streamline the process of DfAM. This will be accomplished by:

- Enhancing and developing the existing design methodology by evaluating existing methods and software commonly used.
- Creating an automation Python framework that connects different design, optimization, and simulation software.
- Testing and evaluating the proposed design methodology by using it to design a component.

1.3 Limitations

This project focuses specifically on Additive Manufacturing, with an emphasis on Laser Powder Bed Fusion (L-PBF), a process in which thermal energy selectively fuses material within a powder bed [1]. Other manufacturing methods are excluded from the scope. Furthermore, the project and the results solely focus on the software available on Linköping University's computers, thus excluding some software that might be available elsewhere.

The project budget is also a limiting factor, allowing only certain licenses within the cost cap and restricting the extent of test printing components.

1.4 Challenges

As mentioned in the limitations chapter, this project will be heavily dependent on the software that is available through Linköping University computers. Therefore, no third-party applications can freely be downloaded for use, and some python libraries are restricted.

Additional challenges lie in technical feasibility and programming skills. At the start of the project, a major uncertainty is the extent to which AM-related software can be integrated through Python scripts and the degree to which these processes can be automated. Furthermore, the complexity of the designed product will influence the scope of the software: a more complex component with additional parameters directly results in a more challenging script to develop.

1.5 Role of Artificial Intelligence in the Project

The artificial intelligence services ChatGPT and Google Gemini were used solely to provide linguistic suggestions, generate illustrative visuals for inspiration and presentations, and assist with LaTeX formatting. All written content in this report is authored by humans, and AI systems were not used as sources of factual information.

2 Organization & Budget

2.1 Project team members & roles

The project group consists of six members, each with a background in mechanical engineering at the master's level. Prior to the start of the project, none of the members had specialized in DfAM. Below is a list of each member and their respective role in the project.

- Kaiz Alhasan – Artificial Intelligence Specialist
- Akshay Awardi – Automation Specialist
- Linus Frostell - Finance Officer & L-PBF Specialist
- Erik Gottfridsson - Project Manager
- Sushil Krishna Vadakkekara Puthan Veedu – Simulation Specialist
- Erik Lindkvist – Document Responsible & Design/Modeling Specialist

2.2 Stakeholders

The project stakeholders include the six-member project team, project supervisor Anton Wiberg, a steering group and potential external partners with an interest in the project. Wiberg, from the Department of Management and Engineering at Linköping University, supports the overall direction of the project, with weekly meetings held to guide and review progress. The steering group is involved less regularly. Their role is to review the project's progress.

Potential external stakeholders could include companies interested in the software and existing AM companies or similar organizations.

2.3 Budget

The project has allocated 3,000 SEK for expenses related to research activities, including car rentals for site visits, software licenses, workshop materials, and similar items related to the project.

3 Expected Results and Learning Objectives

3.1 Expected Results and Deliverables

The project can be divided into three distinct focus areas:

- Literature study, software evaluation and development of optimized design methodology.
- Application of design methodology and design of component.
- Automation through Python scripting or third party software.

These three areas all have several deliverables each that are required to be met for a successful project, and are presented below:

- Literature Study, software evaluation, and development of optimized design methodology:
  - A summary of the problem domain containing: current methods, common software, the current workflow and it's faults.
  - Documented conclusions around suitable software for ease of use, quality and potential automation of the design process, and a compiled list of these software.
  - Proposal of a DfAM methodology optimized for ease of use, quality and automation potential comprised of a clear step-by-step method.
- Application of design methodology and design of component:
  - Definition of requirements for a component consisting of dimensions and loads in order to properly assess DfAM software.
  - Conclusions around the proposed design methodology tested, including reflections and possible improvement areas.
- Automation through Python scripting or third party software:
  - List of software that (feasibly) can be completely or partially automated.
  - Python script or similar that bridges the gap between two or more software.
  - Investigation of third party software and how they can be used to automate the DfAM process.

3.2 Learning objectives

Based on the project description and expected outcomes, four main learning objectives have been identified.

1. The development of skills to use simulation tools (Inspire, nTop etc.) to analyze and optimize the additive manufacturing processes.

2. The improvement of skills in CAD modelling, AM software, automation and simulation to obtain efficient results and research-based solutions.
3. The development of skills in Python for identifying, designing and executing automation tasks in the DfAM process.
4. The improvement of skills in project planning and management by creating and using work breakdown structures, Gantt charts, and preliminary project plans.

4 Theoretical Background

4.1 Design for Additive Manufacturing (DfAM)

Design for Additive Manufacturing (DfAM) is about using additive manufacturing to its full extent. Unlike traditional manufacturing, where the designed components have to fit the limits of traditional machines and tools, DfAM opens up different possibilities. With the process of DfAM, one can make the most of AM's unique strengths. However, one must still respect its limitations.[2]

4.1.1 Key Concepts and Freedoms

DfAM is often divided into opportunistic and restrictive categories.

- Opportunistic DfAM (Design Freedoms) [3] focuses on taking advantage of the capabilities of additive manufacturing which are often difficult or impossible for traditional manufacturing methods:
  - Part Consolidation: Merging multiple components into a single complex part to make assembly easier and possibly reduce costs and weight of the system.[3]
  - Geometric Complexity: Building intricate, or free-form shapes, such as complex internal channel for Fluid Dynamics or Heat Exchange management. [4]
  - Lightweighting/Structural Optimization: Using techniques like topology optimization and lattice structures to place material only where it is absolutely necessary for functionality, and hence reducing mass significantly. [3, 4]
- Restrictive DfAM (Manufacturing Constraints) [3] instead cocuses on the technical limitations of the AM process:
  - Minimum Feature Size: Verifying small features (e.g., walls, holes,struts) will satisfy the resolution limit of the selected AM machine and material. [3]
  - Build Size and Cost: The size of the component, as well as the cost itself can be limiting factors, in particular for high volume work. [2]
  - Anisotropic Properties: The fact that material properties often vary depending on the print direction (anisotropy). [4]

4.1.2 Common DfAM Workflow

The general DfAM workflow is presented below, step by step:

- System Design/Product Planning: Defining the scope, identifying suitable components for AM, defining requirements and understanding interfaces and material selection. [2, 4]

- Component Design: The core of the creative process and optimization work consisting of initial concept creation either often via topology optimization and then modification to ensure functionality. [2, 4]
- Process Design: The creation of a component ready to build including vital steps such as orienting the part, generating support structures, and simulating the process.[2, 4]

4.2 Laser Powder Bed Fusion (L-PBF)

Laser Powder Bed Fusion (L-PBF), also known as Selective Laser Melting (SLM), is one of the most-widely known metal material Additive Manufacturing processes. The process builds components layer by layer using thermal energy to selectively melt local areas of a powder bed of metal particles. [2] The general L-PBF process is presented below:

4.2.1 The L-PBF Process

- Powder Spreading: The process starts with a thin layer of fine metal powder (typically $20-60\ \mu m$ thick for L-PBF) being spread over a heated build platform through blade, wiper or roller. The quality of this layer, known as powder bed density and surface condition is critical to final part quality. [4]
- Laser Scanning/Fusion: The powder particles are selectively melted together by a high energy laser beam, based on the geometry of the current layer as determined from the digital model. This rapid melting and solidification process results into extremely high metallurgical bonds. [4]
- Layer-by-Layer Build: The build platform is lowered by one layer thickness, fresh powder is spread and the process is repeated to create a completed object. The unfused powder is left in the build chamber and acts as a self-supporting material for remaining features. [4]

4.2.2 Critical Manufacturing Aspects

- Thermal Gradients and Residual Stress: The extremely fast heating and cooling cycles in L-PBF processes lead to significant thermal gradient effects. Due to these gradients, different expansion and contraction occurs in the component which builds up large residual stresses. If these stresses are greater than the yield stress of the material, then warping, distortion or cracking can occur. [2, 5]
- Support Structures: For the Laser Powder Bed Fusion (L-PBF), it is indispensable to use additional supports for quality parts and successful builds, serving mainly two functions; mechanical anchorage and thermal management. The support structures act as strong anchors, holding parts in place on the build platform or within underlying levels to prevent overall movement or separation due to high internal residual stresses. At the same time, these features act as a large heat sink to carry thermal energy away from the newly melted

powder and thereby preventing it from being subjected to excessive thermal stresses, which is an important factor in preventing overhang distortion. [4, 5]

- Build Orientation: To achieve manufacturing efficiency and obtain high quality components, the build direction of the parts themselves on the build platform is one of the most important factors. The orientation solely affects the amount of support structures necessary. Therefore, material costs and post-processing times are directly influenced by the position and size of the components. Furthermore, build orientation dictates total height of the component and thus the number of layers and overall build time. In addition to logistics, the selection of build orientation also influences physical characteristics of the part, such as internal residual stress distribution, mechanical properties and surface quality, especially on curved or sloped surfaces. [6]

4.3 Challenges in the current workflow

There are several technical and process barriers between the digital design of a component and a well working functional physical component when working with an L-PBF workflow.

- Iterative Design Loops: Traditional DfAM workflows are linear in that the original digital design has to be manually modified and re-analyzed every time a downstream check such as structural analysis or process simulation flags a failure. This trial-and-error process is inefficient because the designer has to repetitively perform setup operations in different software environments even in the case of slight geometry modifications. [2, 4]
- Organizational and Workflow Challenges: Because the design process is non-linear, component design and manufacturing setup often occur in parallel. This style of workflow can lead to inefficiencies when design iterations are not systematically communicated between the teams, resulting in misaligned manufacturing configurations.
- Support structures: While support structures are vital for stability and thermal energy transfer, they have several downsides:
  - Waste: Support structures lead to material and energy waste that can sometime exceed the volume of the component itself. [5, 6]
  - Post-Processing: Removing support structures once a component is printed is an intensive process that raises the production costs and damages the surface finish and perhaps the geometry of the component. [5, 6]
- Design file interpretation issues: A significant technical challenge is the absence of continuity of data and file formats from software to software:
  - File Errors: The current process involves transferring data between multiple nonintegrated software (e.g., from CAD tool to another simulation tool, then to a slicer). When moving files between software, the files often run into geometry errors, lose data and lose design intent. [2, 4]

  - Remodeling: The rough results produced by topology optimization must be manually remodeled into clean, smooth CAD geometry prior to physical printing. This remodeling step is usually the most time-consuming part during the whole process.

4.4 Process integration and automation

In order to address the aforementioned challenges of complex and multi-stage design processes, Process Integration and Design Optimization (PIDO) philosophy is an emerging concept aimed at bringing together different software in an automated environment and supporting iterative and optimization workflows. [7]

4.4.1 Commercial PIDO Platforms

Tools such as modeFRONTIER and HEEDS are software platforms whose primary aim is to handle complex, Multi-Disciplinary Optimization (MDO) loops. They offer a graphical interface for the integration of different CAD and simulation software. They also serve to define optimization workflows, where parameters are automatically varied and multiple objectives can be addressed at the same time (e.g. minimizing mass, maximizing stiffness, or minimizing support material). [4, 7]

4.4.2 Python-Based Automation Framework

Python serves as the main scripting language to integrate various software tools. This is done by controlling their internal Application Programming Interfaces (APIs), Component Object Model (COM) interfaces, or by executing Command Line Interface (CLI) functions. [8]

Python manages the flow of information by writing to and reading from intermediate files (e.g. STEP, STL or simple TXT/Excel configuration files). [8]

In comparison to commercial Process Integration and Design Optimization (PIDO) platforms, a Python framework is a cost-effective way to provide a high level of functionality without costly licensing for features that may not be necessary for many projects. Although commercial solutions offer out of the box connectivity for common software, a Python driven approach allows full and custom control of specific process steps through custom scripting of tool interactions. After all, the two approaches want to achieve the same result to minimize manual work and speed up design iterations via strong multidisciplinary optimization (MDO) processes that fully realize the true potential of DfAM. [4, 8]

5 Method

5.1 Overall project approach

The overall method in this project can be divided into four distinct phases. Phase 1 is the exploration phase where more knowledge regarding the power bed fusion manufacturing process, its problems, and the DfAM method was gathered through a literature study. The second phase of the project is the analytical phase where the manual software study was performed and all the software were evaluated using the evaluation matrix. In the third phase, called the method development phase, the insights and information gathered from the literature review and the software study was used to produce a DfAM methodology that minimizes materials usage and build-time. The last phase, called the implementation and validation phase, is used to test the proposed DfAM methodology to design a component.

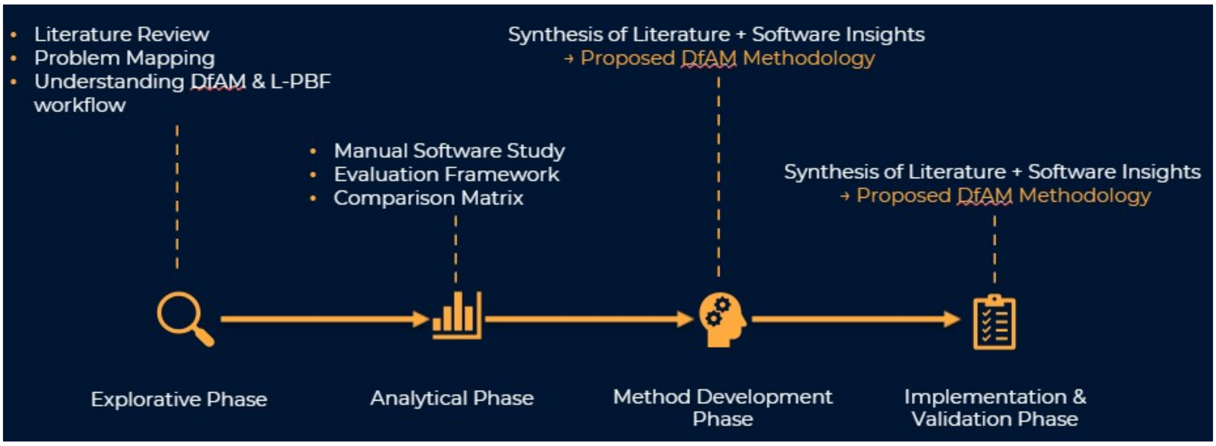


Figure 1: A visualization of the described overall method for the project

5.2 Literature study method

For a literature study to be successful it is imperative that it is performed in a structured and well-defined way, to avoid biases and missing information. The method used for this literature study is one initially developed for software engineering by Kitchenham and Charters [9]. The process of using this method can be summarized as described below. The full scope, keyword list, databases and inclusion criteria can be found in the literature study report [8].

1. Define scope and research questions.
2. Create a review protocol with keywords, databases and inclusion criteria.
3. Systematically search several databases if available, and divide the research questions among group members.
4. Screen the results.
5. Report findings and relate to research questions.

5.3 Component Selection and Standardization

The chosen component to make the evaluation of the software and methodology upon was the so called GE jet engine bracket. This component selection is based on a challenge presented by General Electric in 2013, where contestants could download the bracket to try to optimize its topology as much as possible. The bracket's load case is laid out in the challenge, and includes five loads, a temperature constraint, and a minimum wall thickness. These are:

- Supports in all four of the bracket's bolt holes.
- A force of 8000 lbs vertically.
- A force of 8500 lbs horizontally.
- A force of 9500 lbs 42 degrees from vertical.
- A static torsional load of 5000 lbs-in.
- A service temperature of 24 degrees Celsius.
- A minimum wall thickness of 0.05 in.

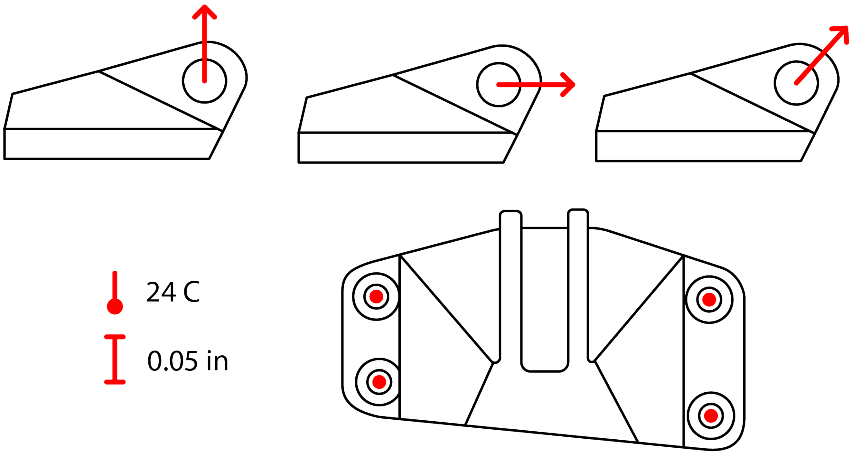


Figure 2: Load cases of the jet engine bracket

5.4 Manual software exploration

In order to be able to evaluate each software sufficiently, each software chosen was evaluated by one or two members of the group. Each software was chosen based on availability on Linköping University's computers and the main functionality or usage of each software. The goal of the evaluation of each software was to understand the basic workflow of each software, how easy they are to use, and the possibility of automation for each tool.

Software	**Main Usage**	**Responsible Author**
SolidWorks	CAD modeling, Analysis, Automation of Process	Akshay
Fusion 360	CAD modeling, Analysis & Topology Optimization	Linus
nTop	Simulation, Analysis & Topology Optimization	Erik L, Kais
Inspire	Simulation, Analysis & Topology Optimization	Erik G, Sushil

Table 1: Main Usage and Division of Responsibility for each Manual Software Testing.

5.5 Software Evaluation

To evaluate and compare the different software in an academic way a comparison matrix with 7 categories was used. The method detailing how each category of evaluation will be presented in this chapter.

5.5.1 Entry-Level Difficulty Evaluation

The Entry-level Difficulty of the different software is inherently a subjective opinion based on the experience of the user. Having used similar software before will, of course, make the learning curve easier. However, regardless of the level of the user, comparing different software is inherently valuable nonetheless, providing insight into which software might be easiest to learn. From the perspective and knowledge base of a 5th year mechanical engineering student, each project member graded the difficulty of software they themselves worked with from 1-5, with 1 being very difficult and 5 being very easy and intuitive.

5.5.2 Accessibility of Documentation Evaluation

The accessibility of documentation category was graded on a scale from 1 to 5. The evaluation of how easy it was to find relevant documentation and tutorials for each software was done subjectively based on each group member's experience when attempting to learn how to use the software. Relevant documentation was defined as any form of document, guide or tutorial that could assist with something in the software in some way.

5.5.3 Simulation Quality Evaluation

The focus of this project does not lie within the evaluation of FEM simulations – however, some FEM simulations were completed during topology optimizations. Therefore, the evaluation of simulation quality was not thoroughly evaluated but rather just subjectively estimated from experience and reliability, on a scale from 1 to 5.

5.5.4 Topology Optimization Quality Evaluation

The topology optimization category was graded on a scale from 1 to 5. To evaluate the topology optimization quality first of all the quality of the result was analyzed in regards to mass reduction and structural integrity, or in other words, if the component actually could support the load without plastic deformation. Another consideration was what functionalities and constraints exist, for example if there are passive/preserved region constraints or overhang limitation constraints present in the software. To aid in this a method called TOPSIS[10] (Technique for Order of Preference by Similarity to Ideal Solution), which in summary entails creating an "ideal" solution and grading how close a solution is to this ideal, was used. The ideal solution for this case is defined as a software that is capable of significant mass reduction without compromising the structural integrity of the component. In addition, it should also have the majority of the following restraint functions:

- Maximum stress constraint.
- Maximum overhang constraint.
- Passive/preserved region(s) constraint.
- Maximizing natural frequencies constraint/objective.
- Symmetry constraints.
- Displacement constraints.

5.5.5 Lattice Structure Generation Evaluation

This section was graded on a scale between 1-5 as most of the other categories. Most important for this grading is defining if it is possible to generate lattice structures in the software or not. If the software has the possibility the function itself is evaluated, for example is it possible to vary the thickness of the lattice? How many different lattice structure cells are there available? Due to stakeholders declaring high interest in this function with regards to lightweight structures, the possibility of lattice generation is deemed highly important.

5.5.6 Manufacturing Simulation Evaluation

The evaluation of manufacturing simulations was also done in a comparative manor, comparing how each software simulates manufacturing. In this case, the software having a slicer tool, build-direction optimization and/or support minimizing functions are all considered great. The manufacturing simulation evaluation category was graded on a scale from 1 to 5.

5.5.7 Slicer Tool Evaluation

The slicer tool category was not graded from 1-5 but rather as a "Yes" or "No". Since slicer tools are abundant and relatively easy to access and export files to, this function was deemed as not essential but a bonus, therefore adding a score of 2 to the total evaluation score.

5.5.8 Automation Feasibility Evaluation

Automation feasibility analysis of each software was done specifically to determine how well each software would integrate into a partly or fully automated DfAM system. This was done at a conceptual level, meaning that fully functional automation systems for each software could have been developed had this been a full-scale project. In addition, some of the software have their own automation tools, such as nTop and nTop Automate, which was also taken into account in the final score, 1-5.

The automation tests worked to assess the level of access offered by official APIs, scripting interfaces, macro facilities, and command-line execution as well as the level at which tasks could be parameterized and performed without user interaction. The other aspect worth considering was the level to which parameters, simulations, and results of designs could be obtained or altered via external scripts written in Python.

Instead of automation performance, what was measured was the relative implementation ease of automation, from an engineering standpoint, which acts as a motivation for the scoring.

5.6 Design Methodology & Component Design

One key deliverable of the project is to define a DfAM methodology based on the software evaluation and literature study, and then using this to design a component. After an overall general design process was defined based on the literature study, the software evaluation guided what software should be used for what step, as well as which functions from said software should be used. Once this methodology had been created it was used to optimize the design of the GE jet engine bracket mentioned in section 5.3.

5.7 Concept for Python automation framework

To address the fragmentation and manual iteration present in current DfAM workflows, a conceptual Python-based automation framework is proposed. The framework follows a modular process chain consisting of CAD modeling, optional AM preparation, simulation, and evaluation or optimization.

Python acts as a connector between the different stages of the workflow by interfacing with software tools through available mechanisms such as APIs, COM interfaces, command-line execution, and file-based data exchange. CAD parameters can be modified programmatically, followed by automated export of geometry files to downstream tools. Simulation software can then be run either through direct scripting interfaces or external process calls, with key results extracted and evaluated.

The evaluation stage enables automated comparison of design alternatives based on predefined performance metrics, such as mass, stress levels, or manufacturability indicators. These results can subsequently be used to update design parameters, enabling iterative loops for design space exploration or optimization.

By structuring the workflow in this manner, the framework aims to reduce manual intervention, improve traceability between design decisions and results, and support scalable and reproducible DfAM processes without being tied to a single software ecosystem.

6 Results

6.1 Summary of literature findings

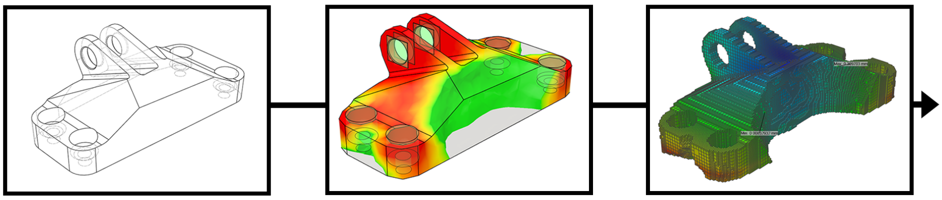
The literature study method is presented earlier in this report resulted in expansive answers to all 10 proposed research questions. The proposed research questions are presented below and the results of these can be found in the literature study report, while the most relevant are presented briefly. [8].

1. How does the powder bed fusion manufacturing process work?
2. What are support structures and what effect does printing orientation have on product characteristics?
3. What are the main problems in the powder bed fusion manufacturing process currently?
4. What does the design process look like from initial idea to final product?
5. Which software are used within the problem domain and which of them are Python compatible?
6. How can Python be used to solve some of the problems of the current process?
7. How does integrating python script with current tools used for DfAM/L-PBF work?
8. What design methods exist currently and how do they work?
9. What type of components are generally manufactured with L-PBF?
10. What does a typical testing standardized component look like in AM?

Research question (4.) is especially interesting in this project from a programming and automation perspective. The summarized answer from the literature report read: "The process can be divided into three quite distinct phases, these being: (i) Initial design definitions where interfaces and loads are defined, (ii) simulation and optimization where the load case is used to generate an "optimal" geometry for the problem. Last is (iii), the process simulation and manufacturing phase, where slicing tools are used to convert the model into appropriate layers, as well as to simulate the required support structures before final printing. A simplified flowchart can be observed in the figure below. [8]

6.2 Manual Process Documentation & Results

The four examined programs (Fusion 360, Inspire, Ntop, and Solidworks) are explored, reviewed, and assigned scores based on the factors presented in Section 5.4; these results are presented in sections 6.2 through 6.8. The resulting scores are presented in 6.8 and serve as a reference to determine each program's relevance in forming a proposed design methodology for designing a component produced using an L-PBF machine.



(i) Initial design (ii) Simulation & optimization (iii) Manufacturing simulation

Figure 3: Visualisation of simplified design process from initial design to ready for print.

6.3 Fusion 360

6.3.1 Introduction

Autodesk Fusion 360 is a program aimed at 3D modeling for CAD, CAM, data management, and more, primarily intended for product design and manufacturing entities [11]. The program includes eight key categories: design, generative design, rendering, animation, simulation, manufacturing, drawing, and electronics. This exploration and evaluation focuses specifically on the simulation and manufacturing features within the program.

6.3.2 Entry-Level Difficulty

The user interface of Fusion 360 is generally considered user-friendly. Tasks are often presented to the user in sequential order, with errors and their causes clearly explained, which helps the user to resolve them. The frequent use of descriptive text and icons helps provide a healthy entry-level experience. For example, the user can hover over features to get explanations of their exact function. While initial navigation within the program can be confusing, it becomes more natural over time.

The downsides of the program's user friendliness lie in details such as surface selection when using certain features. For example, when pardoning faces when generating lattice structure, there is no option to select multiple faces at once. This proves to be time-consuming and frustrating when generating lattice structures for more complex parts. Furthermore, some navigating controls differs from other CAD and CAM software, making it a bit confusing navigating the program at first. When reviewing the program, it is given a score of 4 for entry-level difficulty.

6.3.3 Accessibility of Documentation

Information and discussions about Fusion 360 are widely available on various websites and forums. A benefit is Autodesk's own website, where thorough tutorials and guides exist. YouTube is also a key resource, hosting many tutorials on how to create different solutions within Fusion 360. During the exploration of the simulation and manufacturing extensions, every problem encountered was fixed by troubleshooting online. Because of the ease of troubleshooting and finding content on how to use Fusion, it is given a score of 5.

6.3.4 Topology Optimization Quality & Simulation Quality

In Fusion 360, the Simulation tool includes the *Shape Optimization* function. This function allows the user to optimize a selected shape based on targeted mass reduction while simultaneously displaying the resulting changes to the safety factor. This optimization can be applied generally or for a specific temperature and load case. The process generally consists of 11 steps:

1. Import part file and align to liking
2. Enter simulation mode, select Thermal Stress
3. Under material, change to desired material. In our this Ti-6AI-4v (titanium).
4. Set constraints to features of the bracket that are assumed rigid
5. Define the structural and thermal loads according to the desired load condition
6. Simulate by clicking solve, if constraints or other parameters are overdefined or incorrect, Fusion360 will notify the user about this
7. The simulation is done in a cloud service and takes a few minutes depending on its complexity.
8. Suggestions of an altered mesh is now presented and adjustable depending on desired safety factor. Higher safety factor, the less material removed.
9. The altered mesh is promoted to a separate file.
10. Convert mesh to rigid body.
11. Reiterate stress and thermal test to analyze new shape.

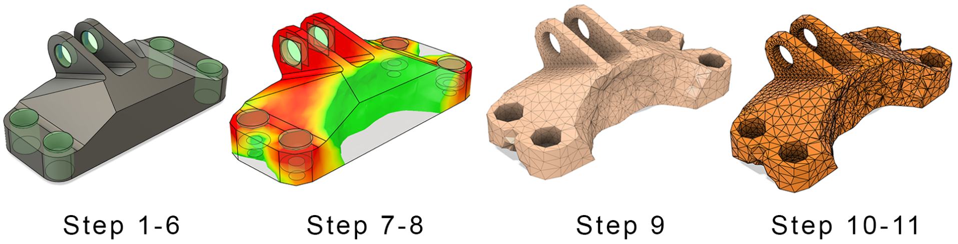


Figure 4: Visualisation of a topology optimization test process in Fusion 360

Fusion 360's topology optimization is useful, primarily due to the simplicity of its execution and the diverse ways it presents results from thermal and stress tests. It is capable of displaying detailed data, including displacement, reaction forces, strain, contact force, temperature, heat flux, and thermal gradients. Furthermore, its ability to simulate predicted deformations caused by applied forces is a powerful tool to help the designer understand the consequence of the stress applied to the product. However, the shape optimization tool itself lacks customization of selectable areas that preferably are reduced most, only allowing the user to pick areas that are to stay the same. The stress test is not possible to accurately perform on a body with lattice structure, which is a big downside for additive manufacturing purposes. Additionally, it defaults to generating undetailed meshes with insufficient subsurface detail which needs to be adjusted before performing the stress test. With the process and results in mind, Fusion's topology optimization quality receives the grade 3 and its simulation quality the grade 3.

6.3.5 Lattice Structure Generation

Fusion 360 has a function named *Volumetric Lattice Generation* which can be applied to any body within the program. The function allows the user to adjust the lattice shape, size, direction, solidity and shell thickness.

1. Import CAD geometry into Fusion.
2. Select volumetric lattice under modify.
3. Customize the shape, proportions, size and scale of the volumetric lattice to liking.
4. Determine solidity, i.e. how hollow the model should be. Numbers near 1 means a solid model, numbers nearing 0 translates to a hollow model.
5. Define offset, creates an outer shell of the lattice structure.

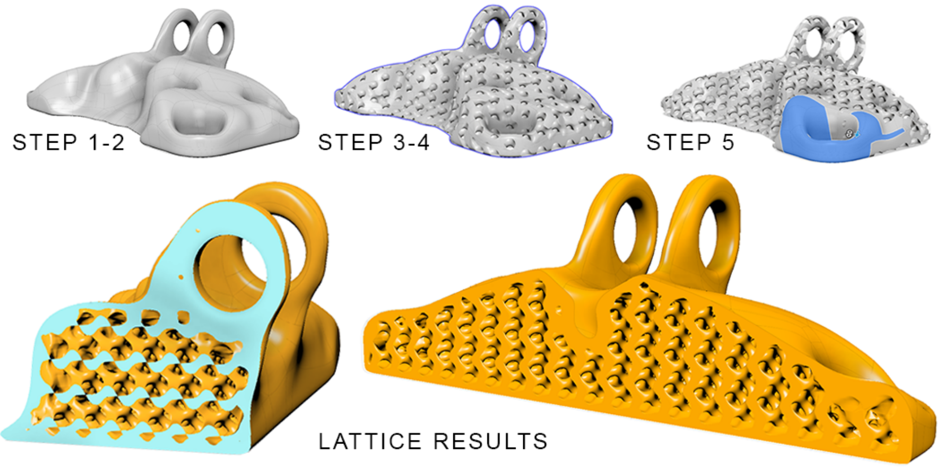


Figure 5: Visualisation of gyroid volumetric lattice generation in Fusion 360. The result images illustrates cross sections of the model.

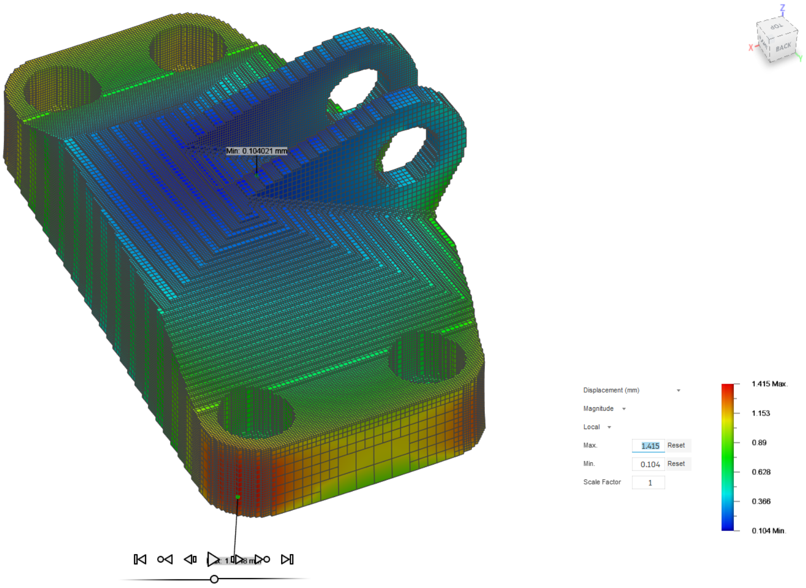
The lattice volume generation feature in Fusion 360 offers high visual quality and relative customization. However, its utility for structurally sound product development is significantly limited by its inability to combine lattice structure generation with shape optimization and its inaccurate stress analysis capabilities for these geometries. Consequently, the feature is unsuitable for designing structurally critical components, such as the jet bracket examined in this study, and therefore receives a grade of 2.

Figure 6: Render of generated lattice structure cross section in ABS plastic.

6.3.6 Manufacturing Simulation Possibilities

In Fusion 360, the manufacturing extension hosts simulation possibilities for plenty of different machines and materials for different manufacturing techniques, among these additive production. Relevant for our project, it is possible to select MPBF (metal powder bed fusion) along with seven other additive methods, e.g. SLS and eBeam. When selecting MPBF, there are approximately 40 machines available to base the manufacturing simulation on. When having picked machine, the following features are possible to control and simulate.

- Layer height
- Sorting type: priority, volume and manual.
- Overhand and support structure, editable manually or automatically
- Bar support
- Additive toolpath simulation
- Pre-checks to find errors with the model
- Studies of the model to identify risk zones and minimum clearances.
- Analyze of temperature, displacerement, structure type etc during manufacturing.
- Creation of machine build file



Displacement (mm)
Magnitude
Local
Max. 1.415 Reset
Min. 0.104 Reset
Scale Factor 1

Figure 7: Slicing results rendered in Fusion 360's manufacturing extension

The pre-checks feature allows for optimization of the printed part's build direction which is positive. While Fusion 360's manufacturing simulation includes most necessary features, it lacks the depth required for the highest score. Specifically, advanced capabilities like in-depth process analysis, granular information reporting, and comprehensive collision avoidance are insufficient. Therefore, Fusion 360 receives a grade of 4 in this category.

6.3.7 Automation feasibility

Fusion 360 has a moderate level of automation because there is a built-in scripting environment that enables Python and JavaScript. On its own, one will be able to automate less-repetitive tasks with scripts or add-ins: modifying geometries, updating parameters, file export, basic setup of some simulations.

Advanced workflows, however-especially those involving topology optimization, lattice generation, and manufacturing simulation-rely heavily on cloud-based services and user interaction, limiting the possibility of fully automated headless execution. Access to simulation results and optimization parameters by scripting is also limited, restricting closed-loop automation.

This therefore renders Fusion 360 to be partially suitable for automation, with most of the tasks being suited for both preprocessing and postprocessing. It receives a rating of 4 for moderate automation feasibility.

6.4 Inspire

6.4.1 Introduction

Inspire is primarily a topology optimization software, designed for additive manufacturing and its preparation. It is not a CAD tool, so it mostly relies on importing already existing CAD geometry and optimizing it for additive manufacturing.

In testing Inspire, the goal was to test its ability to optimize topology and generate a lattice structure. Another general goal was to test the software to see how user-friendly it is. Inspire's optimization workflow from importing the step file, to partitioning the model, to structural analysis and optimization was analyzed and compared to the other software that was tested. In total, four different workflows were tested in Inspire; Topology optimization, structural analysis, lattice structure generation and preparation for printing.

6.4.2 Entry-Level Difficulty

The layout of Inspire reminds one of CAD software like Fusion 360. Functions of the software are laid out in ribbon menus at the top of the screen, like in other similar software. Though CAD features like sketching and extrusion are available in Inspire, we found that these are not as well made or easily understood as in dedicated CAD tools like Fusion 360 or Creo. The tools one needs for topology optimization, like Partitioning and Loads are available in the Structure and Geometry ribbons, which are easy to find and easily understood. Because of these factors, Inspire has been given a score of 5 in the entry-level difficulty category.

6.4.3 Accessibility of Documentation

For Inspire, the available documentation online is plentiful. There exists many tutorials online on how Inspire works from independent creators. However, tutorials from Altair, the company that makes Inspire, are lacking. Therefore, Inspire has been given a score of 4 in this category.

6.4.4 Topology Optimization Quality & Simulation Quality

The topology optimization workflow in Inspire can be described as follows:

1. Import CAD geometry into Inspire.
2. Partition the model with the Partition tool, separating the elements that need to be preserved in the topologically optimized model.
3. Select the material of each body.
4. Input the load case onto the geometry, including supports, external forces and temperature requirements.

5. Define the non-partitioned geometry as a design space.
6. Optimize the model with the Optimization tool. Input the objectives of the optimization before running, for instance mass minimization or stiffness maximization. Also, one must input the safety factor, a measure of how much stress the geometry must be able to handle compared to the input load case. A safety factor of two means the component must be able to handle at least two times the load case.
7. Smoothen the resulting geometry with the PolyNURB tool.
8. Export geometry as a STEP file.

With all the loads applied, one can analyze the structural integrity of the optimized or pre-optimized part. The structural analysis workflow, which in itself is a necessary part of the optimization workflow (since one can not make sure the optimized topology is good enough without it), can in Inspire be described as follows:

1. The loads have to be split up into different load cases (one for each force applied on the same face). In the case of the GE bracket, we need four separate load cases, one for each force applied to the load interface.
2. Run the structural analysis.
3. The results can now be displayed for each separate load case, or as a combination of all.
4. The displacement visualization can reveal areas that need work. The results can also be viewed as the Factor of Safety type, revealing the safety factor in each part of the geometry.

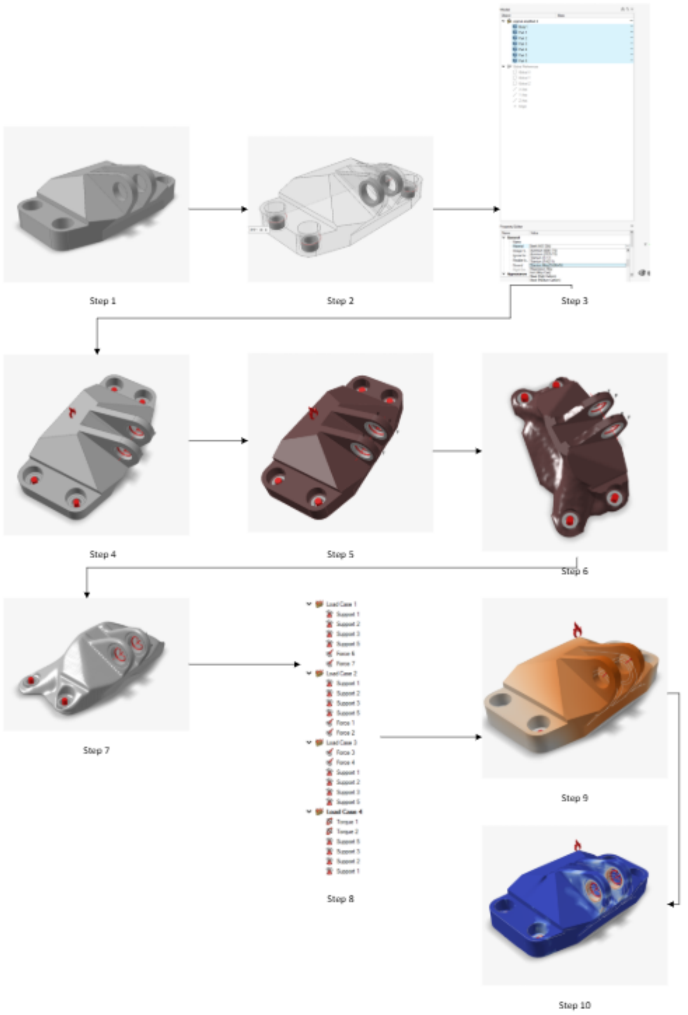


Figure 8: A visualization of each step of the topology optimization and structure analysis process in Inspire.

With the topology optimization and the structural analysis done, it was revealed that the quality of topology optimization in Inspire was very good. Therefore, it was given a score of 5. The simulation quality was given a score of 4.

6.4.5 Lattice Structure Generation

Thirdly, the lattice generation workflow, which can be seen as an extension of the optimization workflow, reducing weight further, can in Inspire be described as follows (a lattice structure can be created both before and after a topology optimization has been made):

1. Go to the implicit modeling tab in the ribbon menu.
2. Clicking on the surface button, a lattice can be created based on unit cell type, whether its supposed to be single or double stranded, coordinate system and density and size of cells.

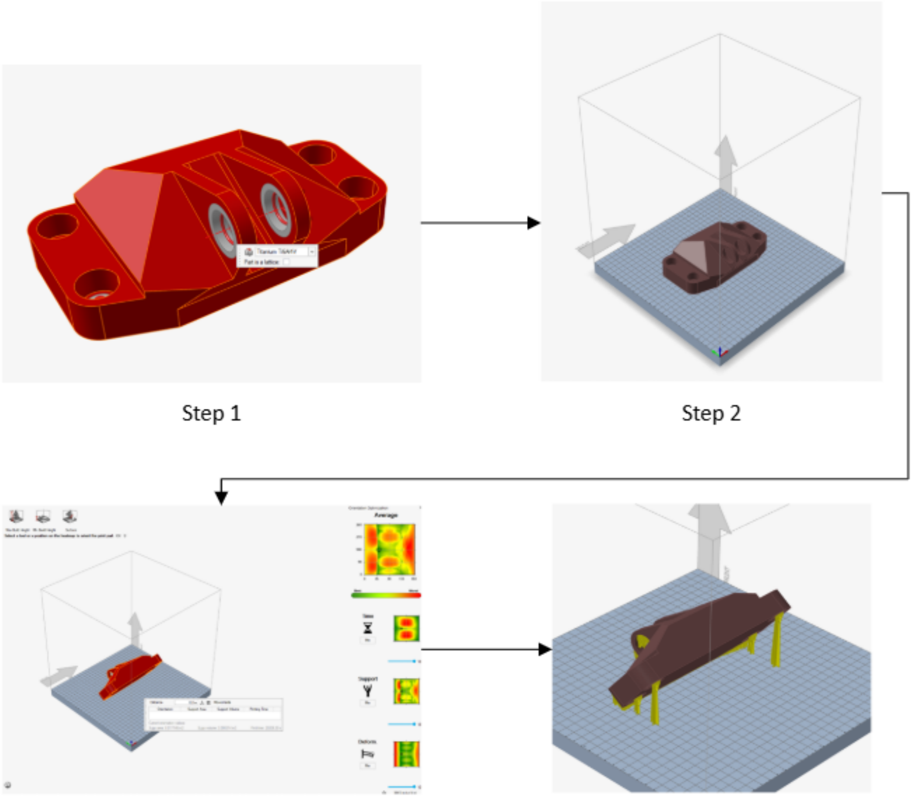
3. After the lattice is generated, the outer body or surface can be made solid again, preserving the aestetics of the part.
4. One should after the lattice structure has been generated make sure that the part can still hold up to loads with a structural analysis.

The lattice generation capability in Inspire leaves a lot to be desired. Though lattices can be created, these can not be created with a goal or a constraint in mind. That is, lattices that are created in components must always be analyzed afterwards, to see if the resulting geometry can still adequately carry the applied loads. Therefore, the lattice generation in Inspire has been given a score of 3.

6.4.6 Manufacturing Simulation Possibilities

Lastly, the preparation for printing process can in Inspire be described as follows:

1. Import geometry into Inspire.
2. Choose one or several parts that one wants to print.
3. Choose the type and size of the printer.
4. Choose the orientation of parts to print. Orientation can be optimized based on time needed, supports needed, or the minimization of deformation, based on a heatmap. One can also orient the part to minimize two or all three categories.
5. Generate support structures for the part. Support structures can be chosen to minimize material intensity or to maximize ease of removal.
6. Export the part for printing.



Step 1

Step 2

Step 3

Step 4

Figure 9: A visualization of each step of the topology optimization and structure analysis process in Inspire.

Inspire's manufacturing simulation works really well. Its build orientation tool is user friendly and intuitive, and its support structure generation is good for minimizing material, time or costs. Therefore, Inspire has been given a score of 4 in this category.

6.4.7 Automation Feasibility

The documentation on automating Altair Inspire with Python is highly lacking. In contrast to Ntop, Inspire is not a list- and block-based program, and more work is done in the viewport. This complicates the automation process in that information can not be taken straight from an excel sheet as it could to a larger extant in Ntop. What could feasibly be done, similarily as in Solidworks, is define simple components with simple dimensions, and load, save and convert files. With this fact in mind, Inspire has been given a score of 2 in this category.

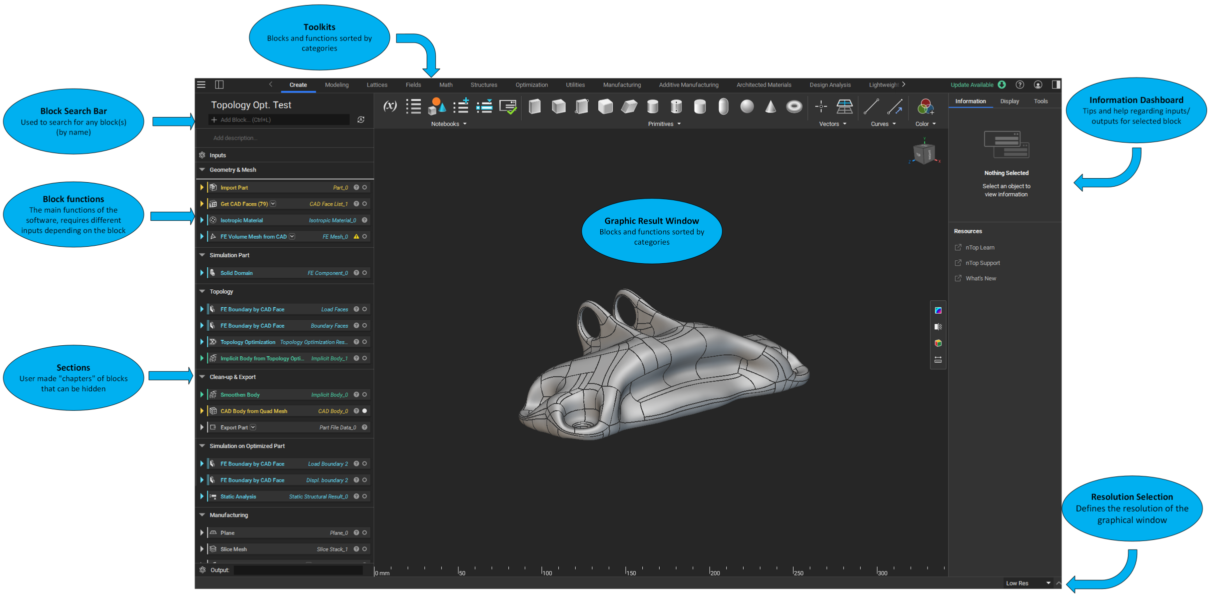
6.5 Ntop

6.5.1 Introduction

Ntop is (mainly) a topology optimization software that also includes functions such as quite advanced lattice generation. It is used by the like of Aerojet Rocketdyne in light weighting endeavors [12] to produce a component for NASA. The goal of testing nTop is to understand how it handles simulations and topology optimization tasks, and how user-friendly its block-based interface is for beginners. The software's lattice generation functions were also tested. The software's workflow, from model import to optimization setup, was analyzed and compared with others in terms of process clarity and automation potential. A special focus was placed on whether nTop allows scripting for integration in a larger automated pipeline.

6.5.2 Entry-level Difficulty

The entry-level difficulty of Ntop was deemed to be quite high due to it feeling very foreign when compared to many (if not all) other software that an engineer is used to. The learning curve is quite steep when trying to adapt to the software's "block" based interface, but after working with it for a good few hours and going through the basic training created by Ntop it started to feel a bit more intuitive. It is worth noting that the error messages and hints in Ntop are quite detailed and very helpful for beginners. Overall the software is quite harsh on beginners and was therefore given a score of 2 in the entry-level difficulty category. Below is a figure depicting the user interface with some explanations.



Toolkits
Blocks and functions sorted by categories

Block Search Bar
Used to search for any block(s) (by name)

Block functions
The main functions of the software, requires different inputs depending on the block

Sections
User made "chapters" of blocks that can be hidden

Graphic Result Window
Blocks and functions sorted by categories

Information Dashboard
Tips and help regarding inputs/ outputs for selected block

Resolution Selection
Defines the resolution of the graphical window

Figure 10: A screenshot from the Ntop software with some notations describing a few functions and windows

6.5.3 Accessibility of Documentation

It would appear that the creators of Ntop are well aware of the steep learning curve of their software, which has led them to creating quite a large amount of tutorials. There exists a tutorial on Ntop's support page for almost every use case imaginable as well as a great "on-ramp" for beginners. With all these tutorials available in the same place, not even considering all the Youtube tutorials, Ntop was given a score of 5 in the accessibility of documentation category.

6.5.4 Simulation Quality

The FE simulation tool in Ntop is quite advanced and detailed, allowing for much control over the mesh size and type. The options and result quality are quite good, but not at the same level as a dedicated simulation software like ANSYS. With this in consideration the simulation tool is more than adequate to support it's use in topology optimization and lattice generation. Ntop was given a score of 4 in the simulation quality category.

6.5.5 Topology Optimization Quality

In the case of Ntop, the topology optimization tool had a plethora of different objectives and constraint functions to utilize. It had no problem reducing the mass by 50% while not altering the (static) interface surface, not exceeding the yield strength of the material and not violating the overhang limitations unless impossible to avoid. All of these functions combined into Ntop receiving a score of 5 in the category topology optimization quality. The workflow used to perform a topology optimization in Ntop can be described as follows:

1. Import CAD geometry and create named variables for load bearing faces
2. Create a finite element volume mesh and face boundaries
3. Input the load case and material parameters into FE simulation model
4. Input simulation model into topology optimization block and define objectives, then run it
5. Smoothen the resulting geometry
6. Create a mesh then use the "quadrangulate" block on said mesh and convert it into a STEP file/CAD geometry

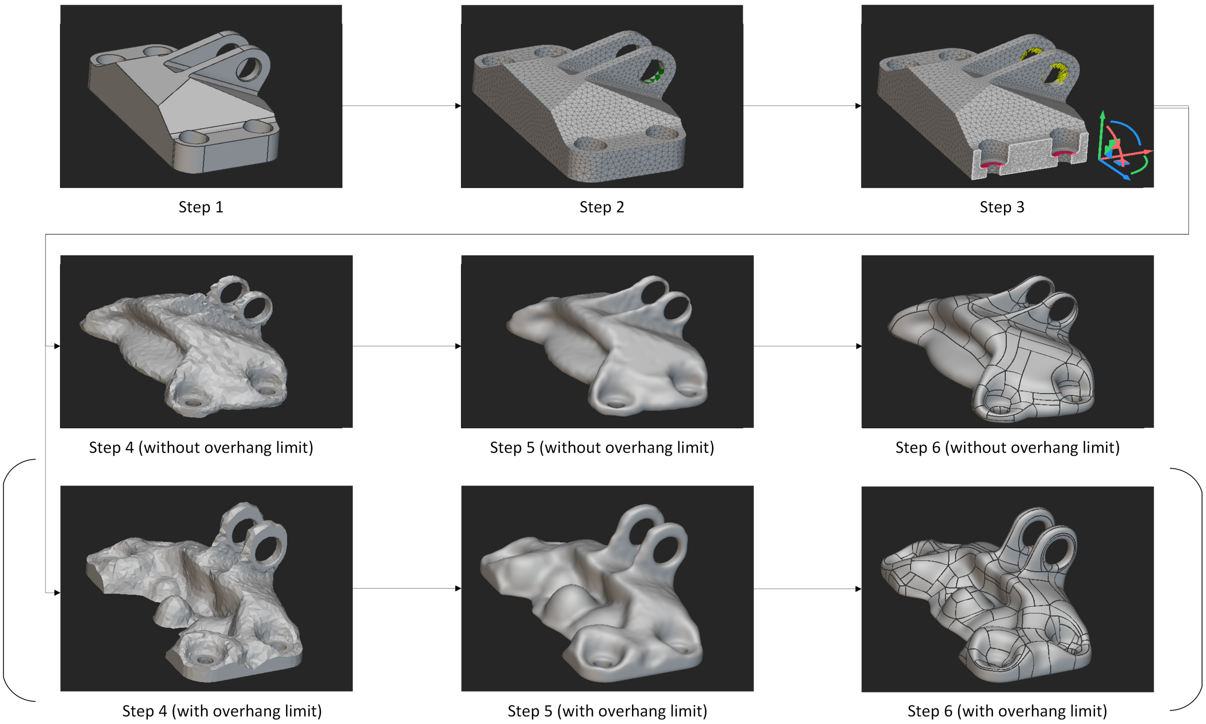


Figure 11: A visualization of each step of the topology optimization process up until the finished part is obtained. Two different constraint cases are shown, the first disregarding overhang and the second having a overhang constraint of 40 degrees. Both runs have a goal of 40 % mass reduction and passive regions at the hole interfaces.

6.5.6 Lattice Structure Generation

The generation of lattice structures is one of the core functionalities of Ntop, allowing for many different patterns, Gyroid and diamond for example, and even lattices with changing "density" based on stress or deformation from a FE analysis. With this said Ntop receives a resounding "Yes" in the lattice structure generation category. The second type of light weighting/mass reduction function that can be utilized is a stress dependent lattice structure. This workflow can be described as follows:

1. Import CAD geometry and create named variables for load bearing faces
2. Create an implicit body
3. Create a finite element volume mesh and face boundaries
4. Input the load case and material parameters into FE simulation model
5. Export the von misses stress (or dislocation) field and use the “range” block to normalize it
6. Create a periodic TPMS cell and input the cells size as approximately double the smallest thickness intended to be used

7. Input the normalized stress field as thickness for the cell
8. Use a Boolean intersect block on the shell and the periodic cell to make the cell map the shape of the desired geometry
9. Use a Boolean union block on the shell and the intersect result to create the final geometry of shell with stress-dependent lattice

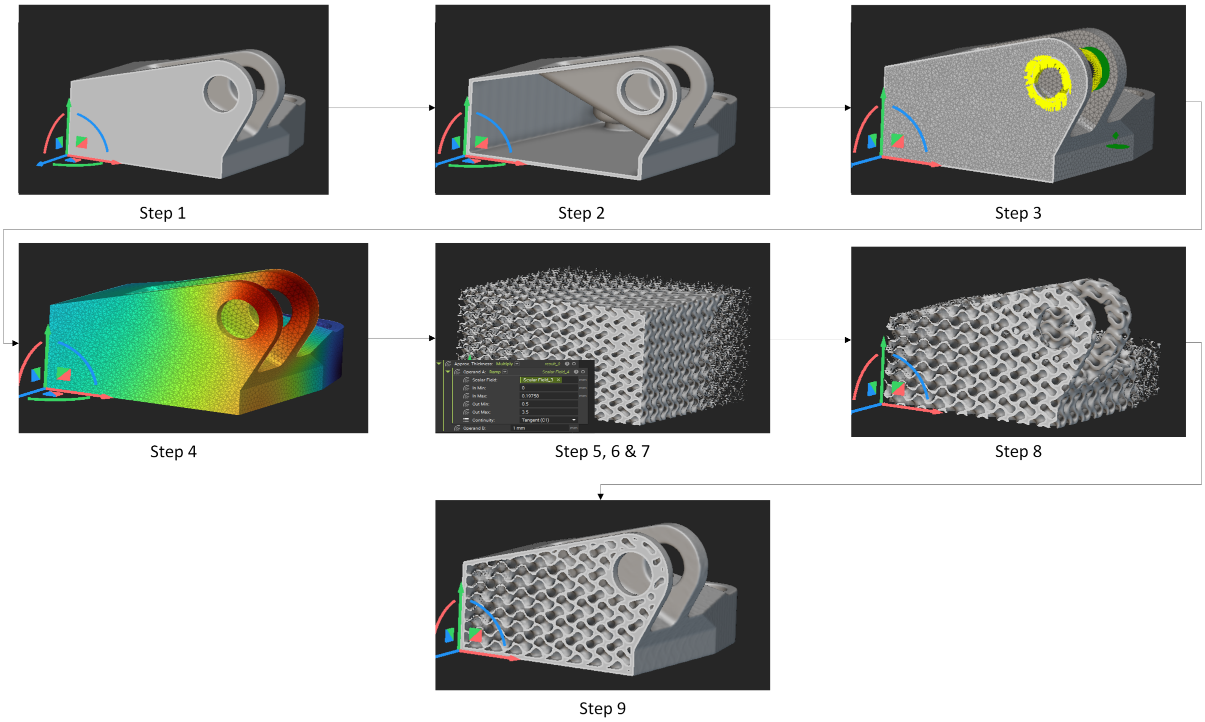


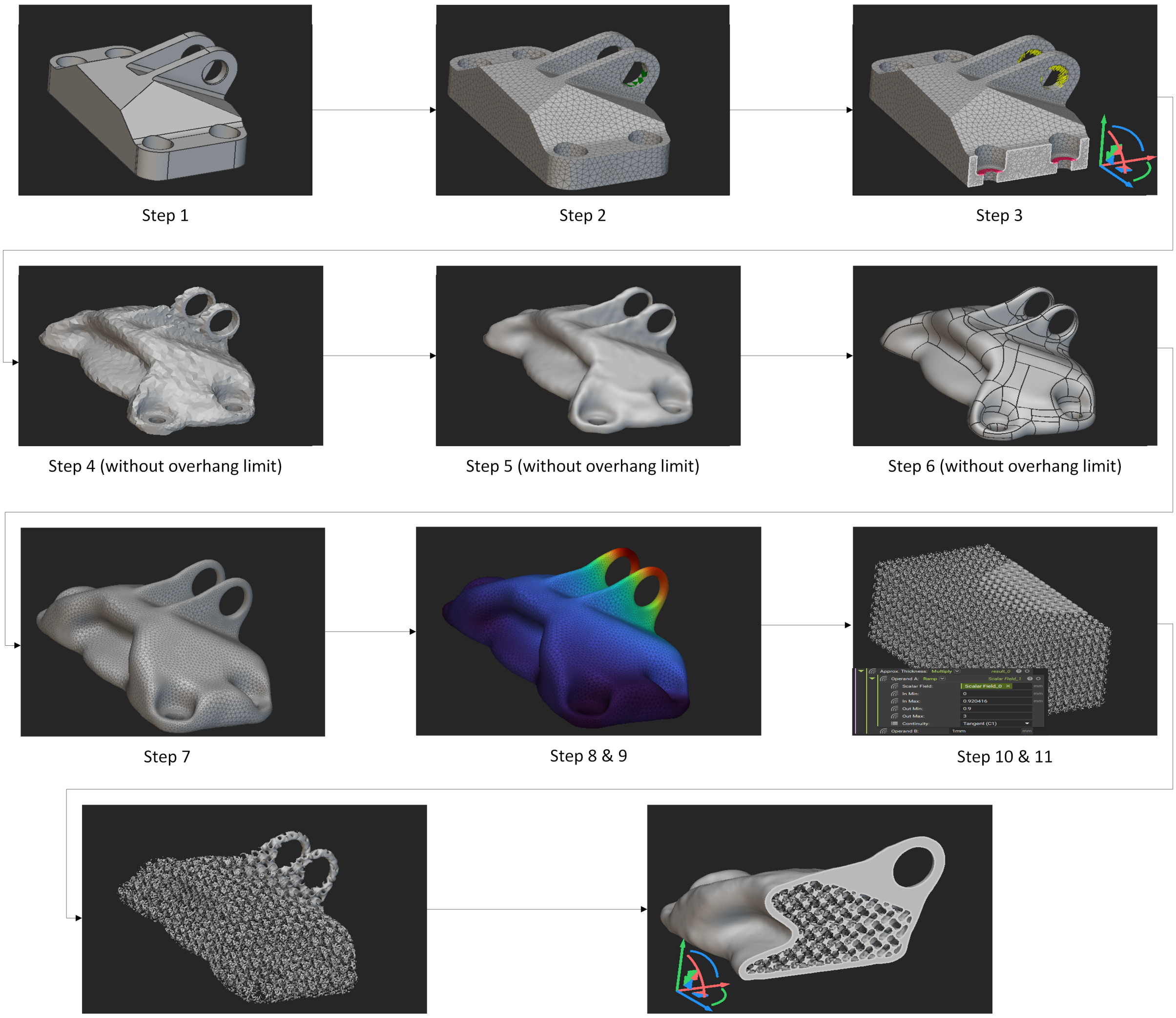
Figure 12: A visualization of each step of the lattice process up until the finished part is obtained.

6.5.7 Combining Topology Optimization & Lattice Structures

The to last type of light weighting/mass reduction function that can be used is a combination of the two methods that are detailed earlier and combines the strength of each method. It can be described as follows:

1. Import CAD geometry and create named variables for load bearing faces
2. Create a finite element volume mesh and face boundaries
3. Input the load case and material parameters into FE simulation model
4. Input simulation model into topology optimization block and define objectives
5. Smoothen the resulting geometry and create a mesh
6. Quadrangulate the mesh and convert it into a STEP file

7. Create a finite element volume mesh and face boundaries
8. Input the load case and material parameters into FE simulation model
9. Export the von misses stress (or dislocation) field and use the “range” block to normalize it
10. Create a periodic TPMS cell and input the cells size as approximately double the smallest thickness intended to be used
11. Input the normalized stress field as thickness for the cell
12. Use a Boolean intersect block on the shell and the periodic cell to make the cell map the shape of the desired geometry
13. Use a Boolean union block on the shell and the intersect result to create the final geometry of shell with stress-dependent lattice



Approx. Thickness: Multiply — result_0
Operand A: Ramp — Scalar Field_1
Scalar Field: Scalar Field_0
In Min: 0
In Max: 0.920416
Out Min: 0.9
Out Max: 3
Continuity: Tangent (C1)
Operand B: 1mm

Figure 13: A visualization of each step of the "combination" process up until the finished part is obtained, This run has a goal of 40 % mass reduction, passive regions at the hole interfaces and no overhang limitation.

6.5.8 Manufacturing Simulation Possibilities

In regards to additive manufacturing simulation tools Ntop has a function that outputs the optimal build direction with objective of minimizing the required support structure for a given part. Within this process a support structure simulation tool also exist as well as a slicer tool that can export directly to the CDF file format. There is also a "minimize height" function that could be useful in some printing cases. Finally the overhang constraint for the topology optimization can also be used to reduce the amount of support structure required to print the part. With all of this in mind Ntop received a score of 5 in the category manufacturing simulation possibilities. A powerful tool is the build-direction optimization for minimizing support volume, this workflow is used at the end of any of the preceding workflows and can be described as follows:

1. Define a plane that will be treated as the build plate
2. Create a "Minimum Support Orientations" block, input the build plane and the smoothened body from the topology optimization (or implicit body created from the lattice and shell). In this step max overhang angle before support is needed and how many candidates to look at can also be defined.
3. Input the transformation list from the step in to a "Manufacturing Support Volume" to obtain a voxel grid describing the support structure volume for each candidate build direction.
4. Use a "Sort" block on the "Transform Object" list output from the step 2 and define the proxy as "Voxel Grid$_0$ > active voxel count" from step 3
5. Export the implicit body from the list with the [0] at the end of it's name to obtain the optimal orientation
6. Optional: Add a new "Manufacturing Support Volume" block and input the build plane and the obtained optimal build direction.
7. Convert the implicit body into a mesh and quadrangulate said mesh
8. Input this mesh into a "Slice Mesh" and export this to CLF with the "Export Slices to CLF" block

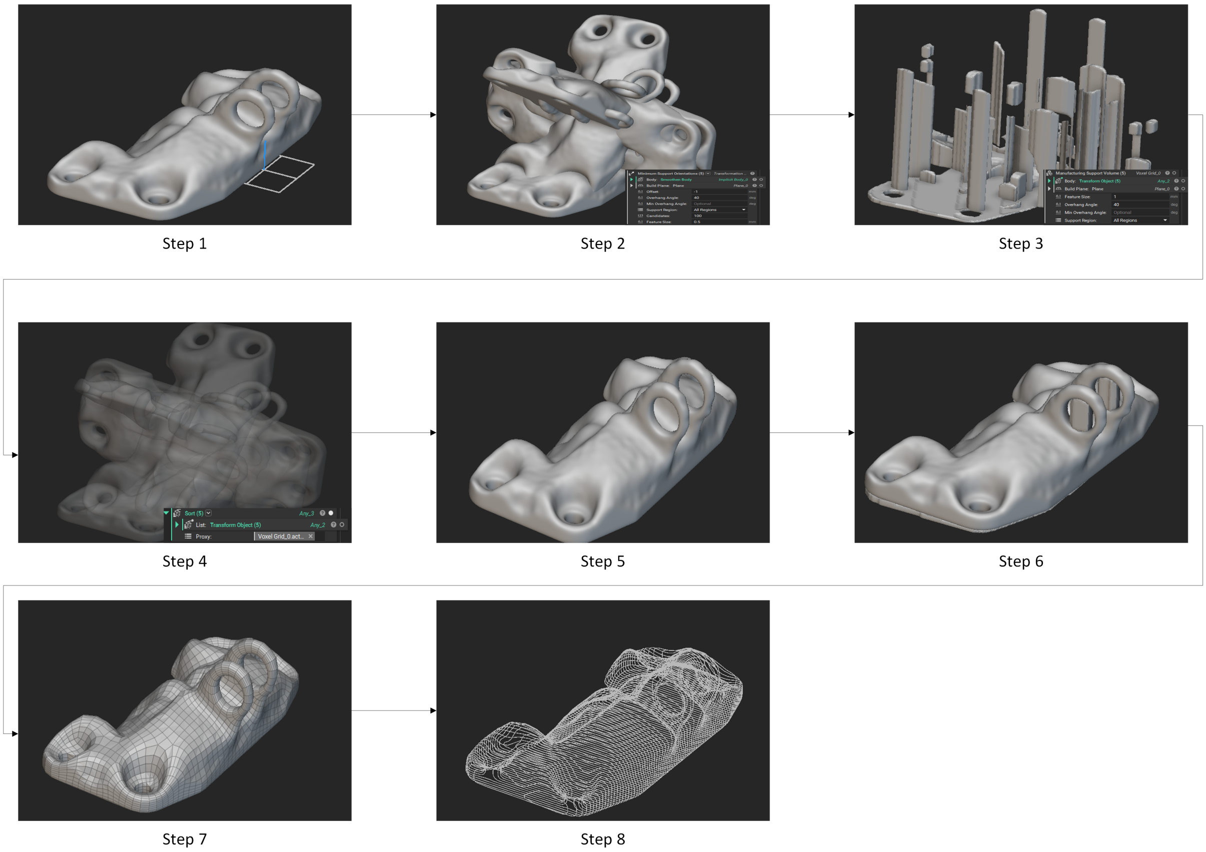


Figure 14: A visualization of each step of the build direction optimization process up until the finished part is obtained, This run was done on the result of a topology optimization with a goal of 40 % mass reduction, passive regions at the hole interfaces and 40 degrees overhang limitation.

6.5.9 Automation Feasibility

It is evident that nTop possesses inherently high feasibility of automation given its fundamentally parameter-driven and block-based architecture. The major steps of geometry import, simulation setup, topology optimization, lattice generation, and manufacturing preparation are explicitly defined through inputs and dependencies. This naturally makes them reproducible and hence well-suited for automation.

Although nTop does not provide a traditional external API for full headless execution, its internal logic allows design parameters, constraints, and objectives to be modified programmatically via structured inputs. This enables batch execution, design space exploration, and integration with external optimization frameworks.

nTop has good alignment from an automation point of view with the goals of reducing manual iteration and enabling DfAM workflows that are scalable. It is thus rated as very suitable for automation within the scope of this project.

6.6 Solidworks

6.6.1 Introduction

SolidWorks is among the leading CAD engineering tools within the industry, popular for its user-friendly interface, advanced modeling functions, and comprehensive integration throughout product development workflows. Dassault System's version of SolidWorks provides parametric and feature-based modeling to help engineers and designers quickly build, modify, and optimize intricate 3D parts, assemblies, and detailed technical drawings.

Over the years, SolidWorks has grown beyond simple CAD modeling into a more advanced set of modules that includes Simulation for structural, thermal, and motion analysis; SolidWorks CAM for manufacturing and toolpath generation; and 3DEXPERIENCE cloud connectivity for collaborative engineering. This solidifies SolidWorks' place in an extremely wide range of industries, from automotive and aerospace to industrial machinery, product design, medical devices, and consumer electronics.

With its extensive documentation, large user community, and strong support for automation through APIs and macros, SolidWorks remains a tool of choice among both students and professional users. Its balance between ease of use and engineering depth makes it an excellent choice for entry-level users while still providing advanced capabilities for high-level engineering applications such as topology optimization, additive manufacturing workflows, and simulation-driven design.

6.6.2 Entry-Level Difficulty

SolidWorks has a rather straightforward learning curve when it comes to part modeling and assembl, and beginners can easily understand how to get started with all the online tutorials available on platforms like YouTube. The difficulties of SolidWorks lie in in other functions such as advanced modeling, CAM, manufacturing, electrical and surface modeling and topology studies. The user interface of SolidWorks is similar to and reminds one of classic CAD-software, with a clean and structured UI with both classic and ribbon menus. Compared to other tools, SolidWorks has more beginner-friendly 3D-modeling and assembly tools which makes SolidWorks easier to learn, and comes with the added benefit of it in a lot of cases being industry standard.

In the context of additive manufacturing, SolidWorks supports topology study, simulation, and manufacturing workflows, making it practical for beginners exploring AM-related topics. Therefore, SolidWorks has been given a score of 4 in this category.

6.6.3 Accessibility of Documentation

Tutorials and help on how to work with SolidWorks are available on YouTube, through community resources like forums and through free and paid learning platforms and courses. The SolidWorks website also has information about the software and its capabilities. Resources are

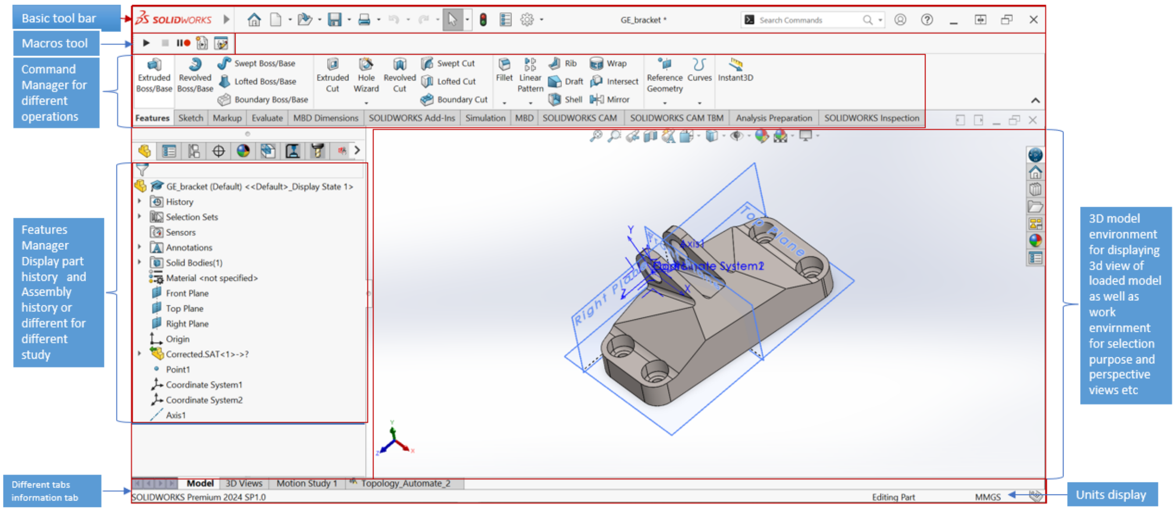


Figure 15: Annotated Overview of the SolidWorks User Interface

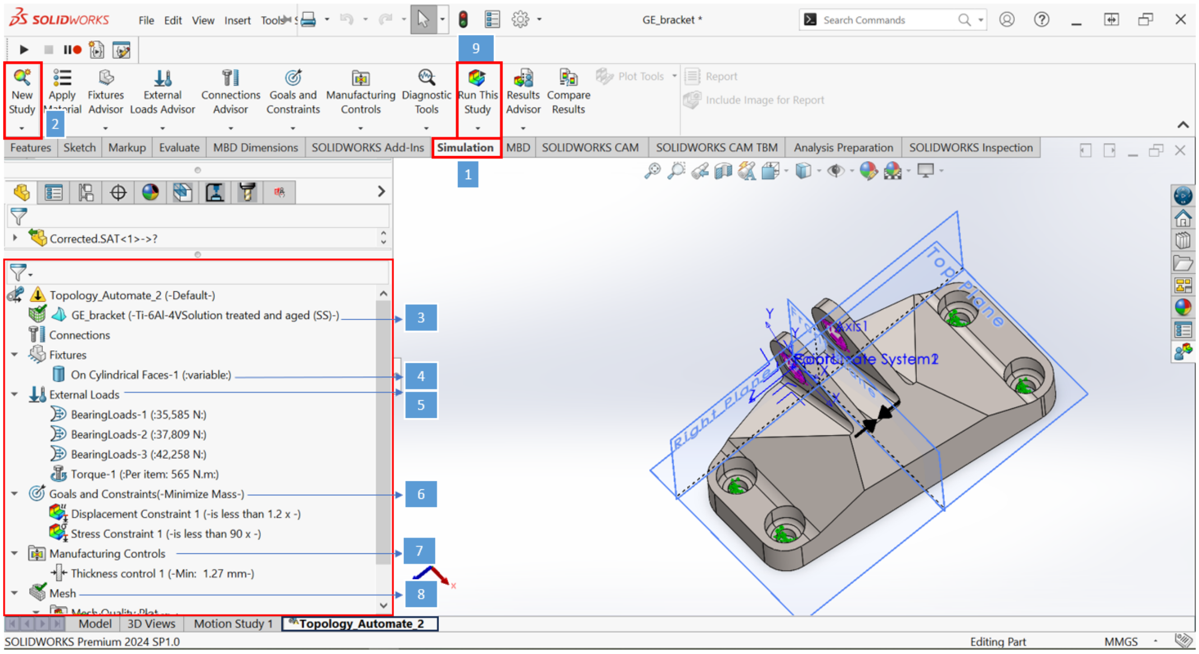
available for for both beginner and advanced users. Therefore, this category has been given a score of 5 for SolidWorks.

6.6.4 Topology Optimization Quality & Simulation Quality

When it comes to Topology Studies in SolidWorks, the options one can choose are self explanatory and easily understood. Topology Optimization can be found under Design Insight in the Simulation ribbon. In order to topologically optimize a component, one first has to select the material, load conditions and boundary conditions, similarly to the other software. The results give a component which is somewhat topologically optimized, as seen in the figure below, but can not be said to have the same quality as either Inspire or Ntop, which is why SolidWorks has been given a lower score in this category. One could discuss whether the results could be improved upon by a more powerful computer, but better results can still be obtained with the same computational power in other available software, as discussed before. Topology optimization and simulation have both been given a score of 3.

The following steps mentioned can be followed to conduct a Topology Study in Solid works:

1. Go to the Simulation ribbon in SolidWorks.
2. Click on the New study tab and Select Topology Study from Design Insight to initiate the Topology study
3. Right click on the model and set the desired material as from the Material Library.
4. Hold Control and Select all the faces which are not to be changed in the topology optimization process.
5. Similarly, hold control and select the faces on which to apply specific loads.



Corrected.SAT<1>->?

Topology_Automate_2 (-Default-)
GE_bracket (-Ti-6Al-4VSolution treated and aged (SS)-)
Connections
Fixtures
On Cylindrical Faces-1 (:variable:)
External Loads
BearingLoads-1 (:35,585 N:)
BearingLoads-2 (:37,809 N:)
BearingLoads-3 (:42,258 N:)
Torque-1 (:Per item: 565 N.m:)
Goals and Constraints(-Minimize Mass-)
Displacement Constraint 1 (-is less than 1.2 x -)
Stress Constraint 1 (-is less than 90 x -)
Manufacturing Controls
Thickness control 1 (-Min: 1.27 mm-)
Mesh

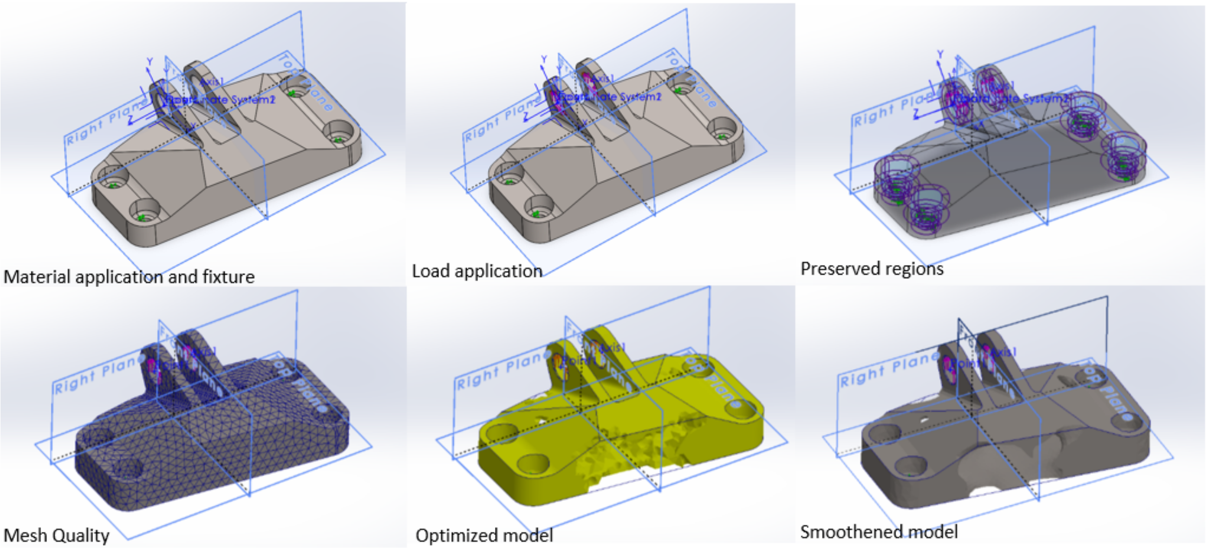
Figure 16: SolidWorks Topology Study Setup

6. The goals and constraints of the process can be set as desired, for example minimizing the mass or displacement level.

7. Manufacturing controls can be set, and faces can be preserved along with Thickness control.

8. Finally, mesh size and type of mesh can be defined and generated.

9. Click on "Run this Study" and wait for the results. The run time depends on the complexity of loads and goals set for the study.


Material application and fixture

Load application

Preserved regions

Mesh Quality

Optimized model

Smoothened model

Figure 17: SolidWorks Topology Study and Quality

6.6.5 Lattice Structure Generation

SolidWorks does not come with lattice structure generation. It can however be manually generated by using patters and sketches using traditional CAD tools, but this is a tedious and time consuming process. This results in a score of 0.

6.6.6 Manufacturing Simulation Possibilities

SolidWorks can simulate the Additive Manufacturing process with the help of paid third party extensions, but does not have Additive Manufacturing simulation capabilities built in. It has been given a score of 3.

6.6.7 Automation Feasibility

SolidWorks comes with CAD Modeling Automation, Assembly Automation, Drawing Automation, Simulation Automation and Workflow Automation capabilities, and has been given a score of 5. However, in this study the focus lies on Simulation Automation, i.e, the Automation of Topology Optimization. Tools utilized in the Automation study were SolidWorks API, VBA Macros, Python Scripts, and Excel and txt files. The SolidWorks Automation pipeline can be designed as per the user's interests, but this study's topology optimization pipeline execution works in the following steps:

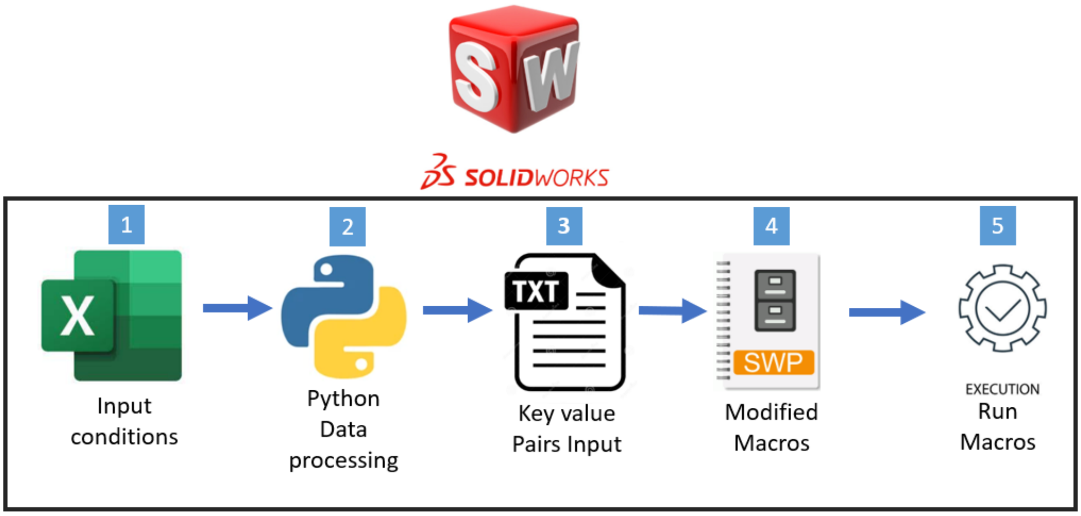


Figure 18: SolidWorks Automated Design Pipeline

1. The user is provided with an input excel sheet to fill in the details of the topology optimization, i.e the name, material, load conditions, goals and constraints, manufacturing controls and mesh configurations.

2. Python scripts process the data in the input excel sheet to provide it to the text file which contains key value pairs as input for modified macros.
3. Key value pairs are overwritten every time the script is run and new values are extracted from the input excel file.
4. The automation process is completed when the macros are executed in SolidWorks, and the given input boundary conditions are applied in the initiated topology study on the component.

6.7 Slicer tool

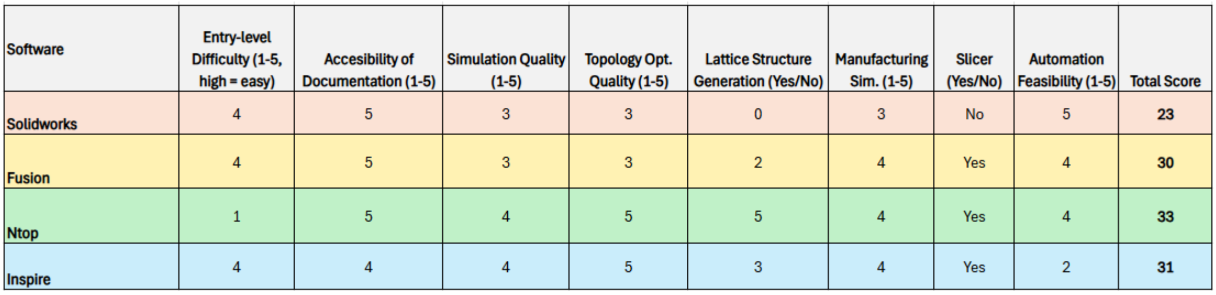
When presenting the slicer tools (or lack there of) there is not much comment to be made, therefore this section will be presented in a table below with a simple "Yes" or "No" indicating if the software has a slicer tool.

Fusion 360	Inspire	Ntop	Solidworks
Yes*	Yes	Yes	No

Table 2: Slicer Tool Table (*Depends on the extension packages installed)

6.8 Summary of Scoring

In Figure 10 the scores of the 4 software in each category is presented, along with the total scores and responsible group members. This was used to guide decisions regarding the new proposed design methodology.



Software	Entry-level Difficulty (1-5, high = easy)	Accesibility of Documentation (1-5)	Simulation Quality (1-5)	Topology Opt. Quality (1-5)	Lattice Structure Generation (Yes/No)	Manufacturing Sim. (1-5)	Slicer (Yes/No)	Automation Feasibility (1-5)	Total Score
Solidworks	4	5	3	3	0	3	No	5	**23**
Fusion	4	5	3	3	2	4	Yes	4	**30**
Ntop	1	5	4	5	5	4	Yes	4	**33**
Inspire	4	4	4	5	3	4	Yes	2	**31**

Figure 19: describing figure of all evaluation scores for each software as well as the total of each software

6.9 Proposed Design Methodology

The proposed design method is compiled of an enumerated list that describes each step of the process of going from base-line CAD model to the printing of a topology optimized shape with stress-dependent lattice structure. This will be presented in two different ways, one in-depth list were software recommendations are included and one "simplified" list that is applicable with any suitable software.

First up is the simplified/general method:

1. Create a CAD model and simplify in any suitable CAD software.
2. Simplify the geometry if complex in any suitable CAD software.
3. Import in to any topology optimization-capable software.
4. Run a topology optimization with passive/constant regions at load faces, fixed faces and bottom as well as a overhang constraint of around 50 degrees.
5. Convert into STL/STEP to be able to run a FE simulation in a suitable FE software
6. Generate the lattice structure based on the FE simulation in a suitable software (most likely the same one used for the Top.Op.).
7. Create a mesh of the component and export it as a "3MF" file if another software is intended to be used (if not, just create a mesh).
8. Use a suitable software to run a build-direction optimization to minimize support structure volume and time while maximizing structural integrity.
9. Export to slicer (or slice in the last used software if possible) and print the part

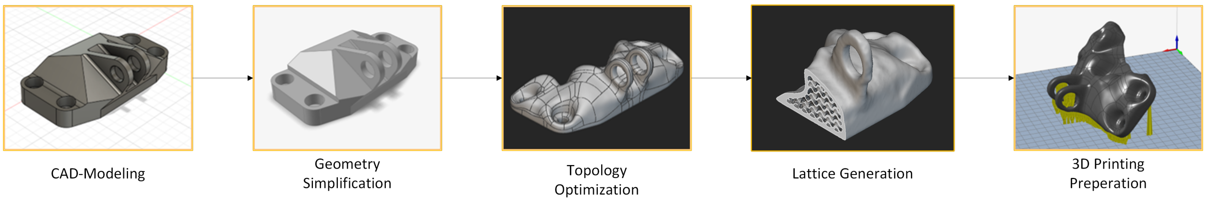


Figure 20: Visualization of the result of each major step in the simplified/general design method

And for the more in-depth method using the software explored during the project:

1. Create a CAD model in Fusion360 or SolidWorks.
2. If needed, simplify its geometry in Inspire.
3. Import the STL (or similar) file into Ntop.
4. Define surfaces for the load case and for passive regions (regions not to be altered by the optimization), as well as a FE volume mesh from CAD body.
5. Run an FE model using the defined surfaces and the desired material properties.
6. Input the FE model results into the topology optimization block and define all desired variables (volume fraction, maximum stress/overhang).
7. Run the topology optimization and smoothen/simplify the result.

8. Create a mesh from the resulting implicit body, then quadrangulate the mesh and convert it into a CAD body.
9. Define faces for the load case for this new CAD body.
10. Run a new FE simulation and export the resulting stress or deformation field, then use the "range" block to normalize it's values.
11. Create a periodic TPMS cell and input the cells size as approximately double the smallest thickness intended to be used.
12. Input the normalized stress field as thickness for the cell.
13. Create a shell of the part then use a Boolean intersect block on the shell and the periodic cell to make the cell map the shape of the desired geometry.
14. Use a Boolean union block on the shell and the intersect result to create the final geometry of shell with stress-dependent lattice.
15. Create a mesh from the implicit body result (Remember: Set min feature size to smaller than the lowest lattice thickness).
16. Export the mesh as a 3MF for use in Inspire.
17. Import the 3MF file into the software Inspire.
18. Define the part to print in Inspire.
19. Define the size and type of printer.
20. Choose the optimal print orientation based on time minimization, deformation minimization or support structure minimization, or a combination of two or three.
21. Generate support structures for the component.
22. Print!

6.10 Component Designed with Proposed Methodology

The component that is achieved as the result of following the proposed design methodology with software recommendations detailed above is a new jet engine bracket. This new bracket has a new, topology optimized, shape and a hollowed out inside filled with a lattice structure that has varying thickness. This change resulted in a 64% reduction in volume, which roughly equals 2.3 kg in the simulated material. The shape is also optimized to minimize the amount of overhang angles present in the part to ease production. This fact in conjunction with the build-direction optimization makes for a very efficient printing process.

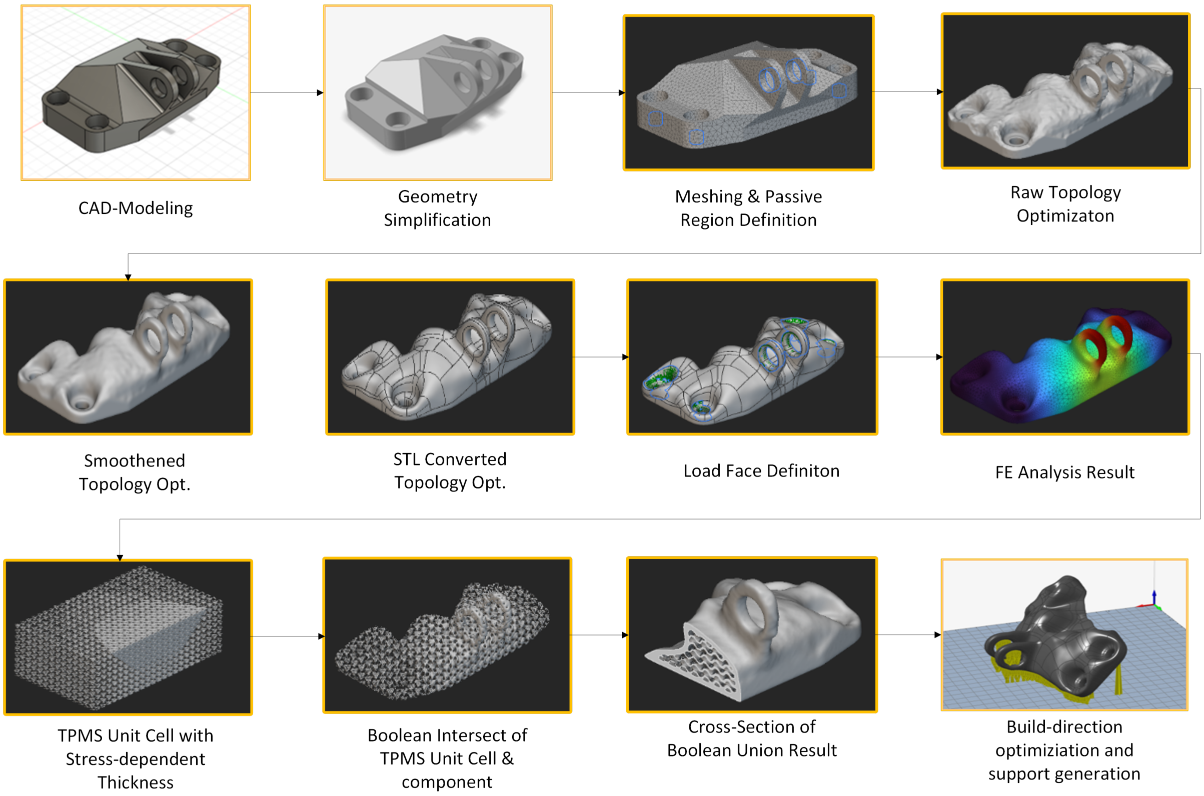


Figure 21: Visualization of the result of each major step in the detailed design method

6.11 Process automation suggestions

In addition to direct automation of individual software tools, the project also explored the use of third-party process integration and design optimization platforms. These tools aim to orchestrate multidisciplinary workflows by connecting CAD, simulation, and optimization software in a unified environment.

As part of this exploration, discussions were held with industry experts and software providers to understand the practical capabilities and limitations of such platforms. One example is pSeven, a commercial process integration and design optimization (PIDO) platform designed to manage automated, data-driven engineering workflows.

While third-party platforms offer powerful optimization and workflow management capabilities, their adoption must be weighed against factors such as licensing cost, setup complexity, and the level of control required over individual process steps.

7 Discussion

7.1 Method discussion

This section will provide discussions regarding the different methods that have been used during this project. The positives and negatives of these methods will be discussed along with

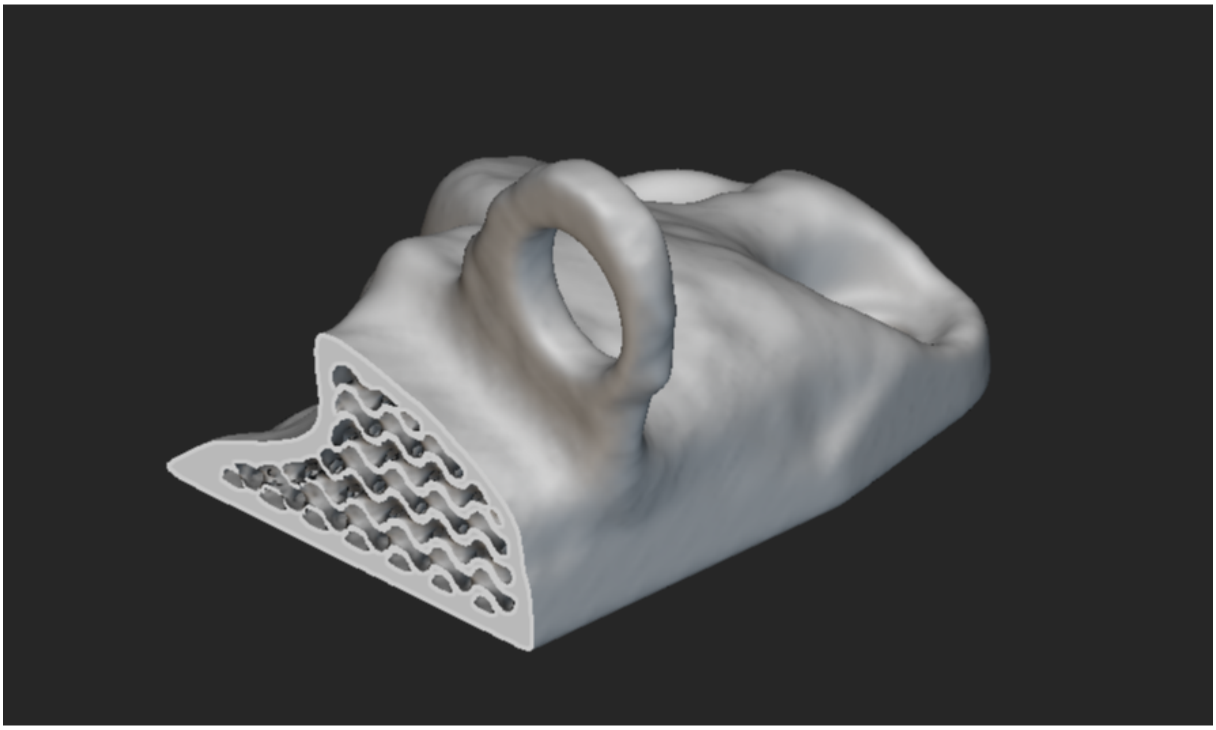


Figure 22: The component cross-section achieved when the proposed method is applied on the GE jet engine bracket.

suggestions as to what could be done differently with the benefit of hindsight.

7.1.1 Literature Study Method

The literature study performed during this project was extensive but mostly focusing on laser powder-bed fusion printing as a process. This study could have been expanded in several different directions to inflate the pool of knowledge that concerns this project. To support the software study more software information could be added to the scope of the literature study. This could lead to new potential software being uncovered as well as possibly finding a second opinion on the already covered software list.

It could also have benefited the project to include other automation possibilities than ones that were Python based, for example p7 and MODEfrontier. This could have put these software on the radar of the project much earlier, which could've led to there being time to test them.

7.1.2 Software Study Method

The software study was done by dividing the group among the four software and evaluating them across 8 categories as described in section 5.5 and 5.6. Another way of doing it could've been to work in two groups of three people exploring two software per group, but this was not done due to the limited amount of value a third person exploring a software was deemed to have. There was also a risk that the depth of understanding for each software would be negatively impacted if every team member was required to explore and learn two different software rather than one.

The evaluations themselves could have been done quite differently as well since the way (most of) the categories were evaluated were mostly based on a subjective evaluation and a comparative discussion. This means that each software group evaluated their own software and then the entire project group discussed the functionalities and perceived quality of said functionalities and adjusted the grading score accordingly. Some categories like "Entry-level Difficulty" and "Accessibility of Documentation" are quite hard to quantify in any other way than at least semi-subjectively, while other categories such as "Simulation Quality" as well as "Topology Optimization Quality" could have been evaluated more objectively. This was not done due the focus of the project being more towards just providing a general workflow/method and the main use of this grading was to help guide us in which tools to use (and not use) for each step in our proposed design method. The actual choice of software done by potential users of the method will most likely not be based on our grading or recommendation, but rather accessibility and pricing of licenses. Had there been more time available it could have been interesting to dig a bit deeper in to the simulation-, topology- and lattice generation quality, alas this was not possible within our time frame.

7.2 Result Discussion

In this section the results of this project will be discussed and evaluated. Some choices made that led to these results will be discussed and motivated as well.

7.2.1 Manual Design Method

The expanded design method that was presented in section 6.9 takes advantage of 3-4 different software to extract the best functionalities and capabilities from each one of the software that were explored in the software study. The design methodology that this paper presents will most likely not be feasible or practical for a "real" use situation like a professional setting, since a company is unlikely to want to get all 4 licenses, which is why the "simplified" methodology that can be used with any software a company or other user may wish to use is also presented. However, in a university setting, the full design methodology that this paper proposes can be used to streamline the topology optimization and lattice generation process.

There could also be alternative versions of the method where only lattice generation or only topology optimization is utilized. There are undoubtedly cases were it would be beneficial to only use one of these light weighting tools but to voice any kind of recommendation more information must be gathered. Due to limited time, this was deemed to be outside the scope of this project.

7.2.2 Automation Framework

Based on the software evaluation and the identified process challenges, a conceptual automation framework is proposed to support partially automated DfAM workflows, as presented in 6.6.8 and 6.11. The framework is intended to connect CAD, simulation, and evaluation tools in a structured and repeatable manner, while allowing flexibility in software selection depending on availability and project requirements. The automation methodology proposed uses excel and text files with python scripts in order to modify input conditions like constraints and loads. Therefore, the automation solution that this paper proposes can theoretically be used not only in topology optimization automation, but in any kind of input-output software relationship.

During the project, the automation capabilities of specialized tools were also explored. nTop Automate was identified as a particularly promising solution due to its native support for parametric, block-based workflows and its suitability for batch execution and design space exploration. However, practical evaluation of nTop Automate was not possible within the scope of this project due to licensing limitations.

In addition, third-party process integration and design optimization platforms such as pSeven were investigated through discussions with industry experts. pSeven provides functionality for orchestrating multidisciplinary workflows and optimization studies across multiple tools, and may therefore be of interest for future work involving large-scale or industrial automation of DfAM processes.

While these third-party solutions were not implemented directly, the insights gained reinforce the relevance of automation and workflow integration as key enablers for more efficient additive manufacturing processes.

8 Conclusions & Possible Future Work

8.1 Conclusions

Before presenting any of the conclusions drawn from this project, the goals presented in the introduction will be presented:

1. Developing a design method based on the evaluation of existing methods and software, like lattice designs and generative design methods
2. Creating a Python framework that connects different design, optimization, and simulation tools
3. Testing and evaluating the created method by using it to design a component

The results of this project, with regard to goal 1, point towards the conclusion that using one singular software for the entire "design-chain" is technically possible (With Fusion360 and SolidWorks) although not advisable. This is due to the CAD-centered tools not performing anywhere near as well as the topology-centered software in the topology optimization and lattice generation. Due to different software having different tools (and quality of tools) it can therefore be concluded that the best design result is achieved using several software. Based on the results of the software study the recommendation of using one CAD-centered software and one Topology-centered software can be done.

Regarding the second goal and based on the proof-of-concept of software automation presented in the project another conclusion could be made. This conclusion being that automation of certain parts of the DfAM design chain, such as simulation setup and topology study execution, can be automated to save time and manual effort. It is however still unclear which particular parts/processes are suitable for this type of automation, since this could not be explored within the time constraints of this project.

In regards to goal 3 it can be concluded from the testing results of the proposed design methodology that the proposed design flow works and does indeed provide a lighter product. The test of the process performed on the jet engine bracket resulted in a 64% reduction in volume (2.3 kg in weight). It is however unknown if this is precisely the optimal process for making this particular product lighter, since many different variables can be changed to slightly alter the result.

8.2 Future work

There are several different directions that further work in this project could go; There could be a greater evaluation work that more thoroughly and methodological evaluates the software. This would be an interesting result due to it being a great guide for which software to get, if resources in a company or project can only cover one or two software.

Another area of research that should be evaluated is how the methodology that this paper proposes would fit into a larger process flow; If the proposed methodology could actually streamline the process in a real-world situation, and let experts work on other, more important tasks.

Another direction future work could take is to continue to work on the automation framework by expand it to include other processes such as lattice generation or build-direction optimization. If the automation framework is expanded the project team would recommend to start with the process of build-direction optimization since this is quite a repetitive process with very little variation, perfect for automation. The automation could also be expanded on in other ways such as exploring the automation tools P7, nTop Automate, and modeFRONTIER rather than creating a new framework.

There could also be further exploration done in the strengths and weaknesses of lattice generation contra topology optimization, since this is not touched upon to any greater extent in this project. This could be used to produce several different, use-case specific, workflows that are more finely tuned to work in a specific set of circumstances.

Acknowledgements

The Project group would like to extend our thanks to Dr. Anton Wiberg for the possibility to perform this project as well as excellent guidance in his pivotal role as supervisor. We would also like to thank the steering group for meaningful and constructive feedback at different points during the project.

References

[1] Igor Yadroitsev, editor. *Fundamentals of Laser Powder Bed Fusion of Metals*. Elsevier, 2021.

[2] F. Dalpadulo and Others. Powder bed fusion integrated product and process design for additive manufacturing: a systematic approach driven by simulation. *The International Journal of Advanced Manufacturing Technology*, 2024.

[3] Stefano Rosso and Fedrico Uriati. An optimization workflow in design for additive manufacturing. 2021. URL `https://www.mdpi.com/journal/applsci`.

[4] Anton Wiberg. Towards design automation for additive manufacturing: A multidisplinary optimization approach. 2019.

[5] Yang Mo Jinlong Su et al. Additive manufacturing by design for support structures: a critical review. *The International Journal of Extreme Manufacturing*, 2025.

[6] Xun Xu Jingchao Jiang et al. Support structures for additive manufacturing: a review. *Journal of Manufacturing and Materials Processing*, 2018.

[7] B. Kitchenham and S. Charters. Guidelines for performing systematic literature reviews in software engineering. Technical report, Keele University, 2007.

[8] Linus Frostell Erik Gottfridsson Sushil Krishna Vadakkekara Puthan Veedu Erik Lindkvist Kais Alhasan, Akshay Rajashekhar Awaradi. Automation in laser powder bed fusion: A literature review of process fundamentals and automation potential. Technical report, Linköping University, 2025.

[9] Stuart M. Charters Barbara Kitchenham. Guidelines for performing systematic literature reviews in software engineering. *Research Gate*, 2007.

[10] Cengiz Kahraman, Sezi Cevik Onar, and Basar Oztaysi. Fuzzy multicriteria decision-making: A literature review. *International Journal of Computational Intelligence Systems*, 8(4):637–666, 2015. doi: 10.1080/18756891.2015.1046325. URL `https://doi.org/10.1080/18756891.2015.1046325`.

[11] Autodesk Fusion 360. What is autodesk fusion?, 2025. URL `https://www.autodesk.com/eu/products/fusion-360/overview`. Accessed: 2025-11-27.

[12] ntop. Aerojet rocketdyne: Low-cost quad thruster, 2023. URL `https://www.ntop.com/resources/case-studies/aerojet-rocketdyne-low-cost-quad-thruster/`. Accessed: 2025-11-25.