Chair for Computer Science 10 (Media Computing and Human-Computer Interaction)

RWTH AACHEN UNIVERSITY

Fabric Faces: Printer-Aware Foldable Textile Structures

Bachelor's Thesis
submitted to the
Media Computing Group
Prof. Dr. Jan Borchers
Computer Science Department
RWTH Aachen University

by
Robert Rudnev

Thesis advisor:
Prof. Dr. Jan Borchers

Second examiner:
Prof. Dr.-Ing. Ulrik Schroeder

Registration date: 04.02.2022
Submission date: 07.06.2022

Zentrales Prüfungsamt/Central Examination Office

RWTH AACHEN UNIVERSITY

Eidesstattliche Versicherung
Statutory Declaration in Lieu of an Oath

___________________________ ___________________________

Name, Vorname/Last Name, First Name Matrikelnummer (freiwillige Angabe)
Matriculation No. (optional)

Ich versichere hiermit an Eides Statt, dass ich die vorliegende Arbeit/Bachelorarbeit/ Masterarbeit* mit dem Titel

I hereby declare in lieu of an oath that I have completed the present paper/Bachelor thesis/Master thesis* entitled

____

____

____

selbstständig und ohne unzulässige fremde Hilfe (insbes. akademisches Ghostwriting) erbracht habe. Ich habe keine anderen als die angegebenen Quellen und Hilfsmittel benutzt. Für den Fall, dass die Arbeit zusätzlich auf einem Datenträger eingereicht wird, erkläre ich, dass die schriftliche und die elektronische Form vollständig übereinstimmen. Die Arbeit hat in gleicher oder ähnlicher Form noch keiner Prüfungsbehörde vorgelegen.

independently and without illegitimate assistance from third parties (such as academic ghostwriters). I have used no other than the specified sources and aids. In case that the thesis is additionally submitted in an electronic format, I declare that the written and electronic versions are fully identical. The thesis has not been submitted to any examination body in this, or similar, form.

___________________________ ___________________________

Ort, Datum/City, Date Unterschrift/Signature

*Nichtzutreffendes bitte streichen

*Please delete as appropriate

Belehrung:
Official Notification:

§ 156 StGB: Falsche Versicherung an Eides Statt

Wer vor einer zur Abnahme einer Versicherung an Eides Statt zuständigen Behörde eine solche Versicherung falsch abgibt oder unter Berufung auf eine solche Versicherung falsch aussagt, wird mit Freiheitsstrafe bis zu drei Jahren oder mit Geldstrafe bestraft.

Para. 156 StGB (German Criminal Code): False Statutory Declarations

Whoever before a public authority competent to administer statutory declarations falsely makes such a declaration or falsely testifies while referring to such a declaration shall be liable to imprisonment not exceeding three years or a fine.

§ 161 StGB: Fahrlässiger Falscheid; fahrlässige falsche Versicherung an Eides Statt

(1) Wenn eine der in den §§ 154 bis 156 bezeichneten Handlungen aus Fahrlässigkeit begangen worden ist, so tritt Freiheitsstrafe bis zu einem Jahr oder Geldstrafe ein.

(2) Straflosigkeit tritt ein, wenn der Täter die falsche Angabe rechtzeitig berichtigt. Die Vorschriften des § 158 Abs. 2 und 3 gelten entsprechend.

Para. 161 StGB (German Criminal Code): False Statutory Declarations Due to Negligence

(1) If a person commits one of the offences listed in sections 154 through 156 negligently the penalty shall be imprisonment not exceeding one year or a fine.

(2) The offender shall be exempt from liability if he or she corrects their false testimony in time. The provisions of section 158 (2) and (3) shall apply accordingly.

Die vorstehende Belehrung habe ich zur Kenntnis genommen:

I have read and understood the above official notification:

___________________________ ___________________________

Ort, Datum/City, Date Unterschrift/Signature

Contents

List of Figures

Abstract

Fabric Faces is an alternative way to 3D-print object, by ungolding an object and printing it on a fabric. This structure of the unfolded object can be folded and clipped back into the original form. This can speed up the 3d-printing process, but can also integrate different surface materials with different properties into the print. This thesis will improve accessibility of *Fabric Faces* by integrating it into *Cura* and expand on it by developing an interlock system for the fabric on which the *Fabric Faces* structure is printed on. Furthermore, we will test material use, print time and impact resistance of *Fabric Faces* and compare this to a standard approach to 3d-printing.

Überblick

Fabric Faces is eine alternative Art und Weise zu 3D-drucken, durch auffalten des Objekts und des anschließenden druckens auf Stoff. Die Struktur des aufgefalteten Objekts kann danach wieder zu ihrer ursprünglichen Form zusammengefaltet und geklippt werden. Dies kann nicht nur den Druckprozess verschnelleren, sondern auch neue Oberflächen mit verschiedenen Eigenschaften in den Druckprozess Integrieren. Diese Arbeit wird durch das Integrieren von *Fabric Faces* in *Cura* rein, die Zugänglichkeit davon verbessern. Zusätzlich wird ein Mechanismus entwickelt der den Stoff, auf dem die *Fabric Faces* Struktur gedruckt werden soll, fixiert. Des Weiteren werden wir Tests durchführuen, die Materialverbrauch, Druckzeit und Stoßfestigkeit von *Fabric Faces* testen und diese dann zu einem Standardansatz des 3d-druckens vergleichen.

Acknowledgements

I would like to thank Prof. Dr. Jan Borchers first for allowing me to write my bachelor thesis at his chair.

I would also like to thank Prof. Dr. Ulrik Schroeder for being my second examiner.

For his excellent support and guidance, I would also like to thank my supervisor Adrian Wagner.

Next I would like to thank René Schäfer for his explanations and introductions to the machines in the FabLab and in extension, i would also like to thank the Hiwi's working at, and maintaining the FabLab.

Furthermore, I would like to thank my family and friends for supporting me for the last few months and years of studying.

Conventions

Throughout this thesis we use the following conventions.

Text conventions

Definitions of technical terms or short excursus are set off in coloured boxes.

EXCURSUS:
Excursus are detailed discussions of a particular point in a book, usually in an appendix, or digressions in a written text.

Definition:
Excursus

Names of concepts or programmes are written in *italic*

Source code and implementation symbols are written in typewriter-style text.

```
myClass
```

The whole thesis is written in american english.

Chapter 1

Introduction

3D-Printing , especially *FFF* (fused filament fabrication [CUBTO+19]) *3D-Printing* (commonly known as filament printing), is getting more accessible in personal and industrial spaces for example to create prototypes or low volume spezialized parts.

The big advantage of *3D-Printing* is the possibility to create objects with shapes which would be difficult or impossible to produce with casting or subtractive manufacturing techniques like *CNC-Milling*. And without the need of creating new manufacturing equipment like molds to produce a new variation of an object, *3D-Printing* allows for faster prototyping with lower turnaround time between variations [IG15]. One disadvantage of *FFF 3D-Printers* is the low print speed and the bad physical scalability of prints.

3D-Printers are great at low quantity production runs, but hard to scale

Fabric Faces is a new approach to *FFF 3D-Printing* for rapid prototyping and integration of new surface materials. With *Fabric Faces* an object is reduced to its wireframe (or skeleton) and unfolded to be printed flat on the print bed (fig. 1.1). The walls will be left empty and will be replaced by a fabric material on which this unfolded wireframe could be printed on. The resulting model is comparable to a foldable paper model, which can be folded to its intended shape (fig. 1.2). Connectors are also generated to make the sides clip together.

Fabric Faces is an unfolded wireframe printed on a fabric

Fabric Faces is a *Blender* plugin, which handles the unfolding procedure

Currently *Fabric Faces* can be accessed through a *Blender* plugin, there parameters can be set to adjust the width of the skeleton structure and the size and type of the connectors and a preview will be generated. This preview will show the generated skeleton and the placements of all connector pieces. If the object could not be unfolded in one single structure, further skeletons will be generated and shown next to each other. After the skeleton structures where generated, they can be exported as a *.stl* file (one commonly used file format for 3d objects), and imported to a slicer to prepare them for printing.

First goal is to integrate *Fabric Faces* into a slicer

One goal of this work will be to extend *Fabric Faces* to increase the accessibility by reducing the number of steps a user has to take, to get from an initial model, to a ready to print gcode file of the *Fabric Faces* structure. This will be archived by integrating *Fabric Faces* into a slicer, which will use this *Blender* plugin in the backend. This slicer plugin will handle the configuration of the *Fabric Faces* generator (to ensure the generated structure will be printable), will import the generated structure into the slicer itself will modify the created gcode file, to ensure the part will be printable on a fabric.

An interlock system for the fabric will be developed

Another feature will be the development of an interlock system. Printing on fabric is a major feature of *Fabric Faces* , so an easy to use interlock system will improve accessibility by providing a solution to fix the fabric in place, without the need of modifying the *3D-Printer* .

Second goal is to conduct a technical study and compare *Fabric Faces* to standard prints

Another goal of this work will be a technical analysis of *Fabric Faces* to determine its technical properties like strength print time and material use by comparing it to a standard approach of *3D-Printing* with different slicer settings to determine in which circumstances *Fabric Faces* could be beneficial form a point of rapid prototyping, overall print time saving or more efficient material use.

In the beginning of this thesis, we will present selected related work to show the current state of Fabric Faces and compare it to our expected work flow. Then we will go through the development of the slicer plugin and the changes that has to be made to the *Blender* plugin . At the end we will conduct a two parted technical study to evaluate the technical properties of *Fabric Faces* and discuss our results.

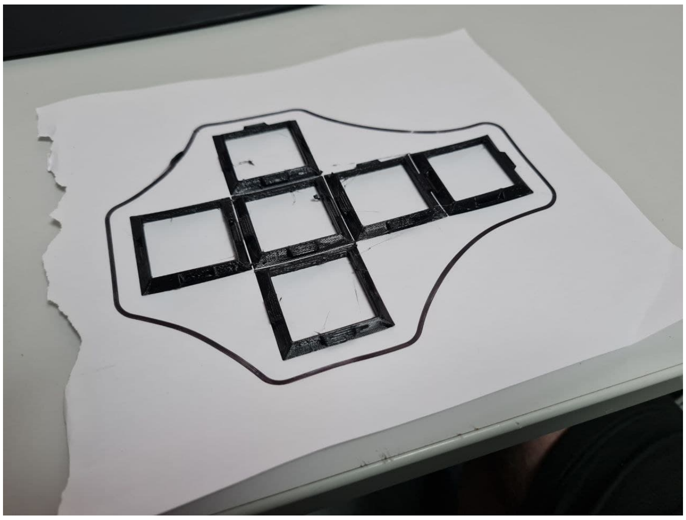

Figure 1.1: Unfolded *Fabric Faces* structure printed on paper

Figure 1.2: *Fabric Faces* structure folded back into its intended form

Chapter 2

Related work

The basic idea of *Fabric Faces* is the creation of a 3D object from a 2D shape with the utilization of a base fabric. In this chapter we will present related approaches to similar problems, focusing on the workflow aspect and integration of different systems.

basic idea of *Fabric Faces* is the unfolding of an object

We will start by presenting a similar approach to *Fabric Faces* to create 3D models out of folded 2D shapes. Next we will investigate an approach of combining *3D-Printing* with an additional system, namely electrospinning, to get different surface finishes and properties. Furthermore, we will examine research, where *3D-Prints* are substituted by laser cut plates.

We will present three distinct papers, first *Kyub* , second *Desktop Electrospinning* and last *Platener* as related work

At the end we will show how this work compares to our own work, and where they differ.

2.1 Clipped shapes creating 3d objects

To illustrate related work on the unfolding of 3D objects into 2D shapes and the reassembly of it, we first present the interactive editing system *Kyub* by Patrick Baudisch et. al. [BSK^{+}19]. This systems allows for the creation of objects in 3D, which will be unfolded into plates designed to be laser cut and reassembled.

Kyub utilizes laser cutters to create 3D structures out of 2D plates

Their idea is to utilize the high speed of laser cutters, compared to the rather low speed of other rapid prototyping techniques like *3D-Printing* and create plates which could be cut and clipped securely together. To archive this they created a construction system based on closed box structures they called *boxels*. By assembling the 3D model with those boxels (fig. 2.1), it remains a box like shape, and will be easily assemblable after it is been cut out.

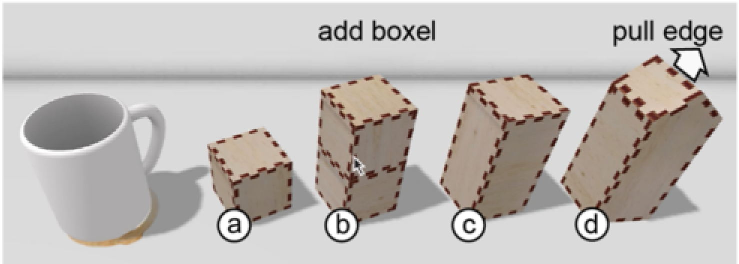

Figure 2.1: Object creation out of boxels, picture taken from [BSK+19]

On export, *Kyub* will automatically break down the model into plates, which are then layed onto a sheet. Each sheet will be exported as a separate vector graphics file.

All in all, *Kyub* is providing a full system in which the user can create 3D models and export them in a ready to cut format, which enhances accessibility.

2.2 Archiving different surface properties with electrospinning

While *Fabric Faces* archives different surface finishes and properties, by printing the *Fabric Faces* structure onto a fabric, *Desktop Electrospinning* by Michael L. Rivera et. al. [RH19] utilizes a new type of *3D-Printer* . This printer combines a classical plastic print with melt electrospinning .

With electrospinning, different surface finishes can be created

By applying high electrostatic potential to a molten or dissolved polymer, thin electrospun fibers are created. By varying the federate, temperature, infill density or electrostatic spin different surface finishes with different properties can be created (fig. 2.2).

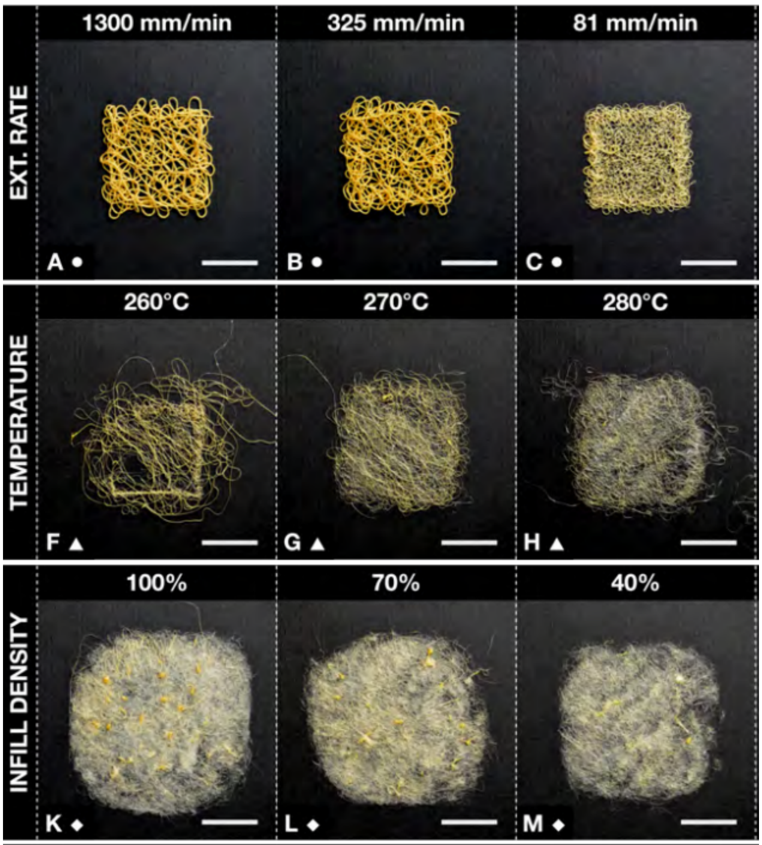

Figure 2.2: Series of electrospun tests, picture taken from [BSK+19]

This kind of integration of different surface properties, while having the potential to create everything on one machine, is not easily accessed due to the requirement of a modified *3D-Printer* and is therefore not ideal for broad adoption at this point in time.

2.3 Substituting *3D-Prints* with laser cutters

Like *Fabric Faces* , which uses fabric to subsidize wall material of a 3D-print, *Platener* by Dustin Beyer. et. al [BGM+15] provides a system for intermediate subsidization of 3D-prints with laser cut parts (fig. 2.3). By specifying a fidelity-

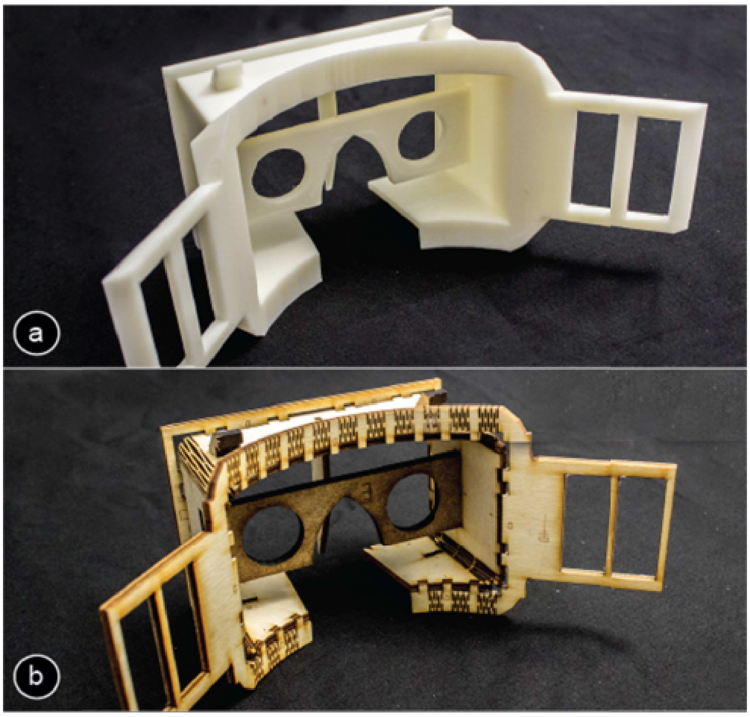

Figure 2.3: *Platener* example for substitution of 3D-printed parts, picture taken from [BGM+15]

Platener approximates a 3D printable part with 2D shapes

speed trade of, *Platener* will try to approximate the 3D object with 2D shapes while trying to preserver the requested amount of fidelity. *Platener* can also handle curved surfaces, by using small plates connected by joints or with one long plate by assuming the part could bended.

2.4 Relation to this work

As we have shown, each of these projects are providing an all in one solution, to access the system they developed. *Fabric Faces* does not have such a system yet. To use *Fabric Faces* you have to use at least two separate programs, *Blender* and a slicer. We want to integrate *Fabric Faces* into a slicer to improve the accessibility. We also plan to allow for integration of other systems like laser cutters, like shown in 2.3 to, to support preprocessing steps.

Chapter 3

Fabric Faces Cura Integration

To make *Fabric Faces* more accessible, we have to reduce the number of steps and complexity of the process. We will integrate *Fabric Faces* into one of the most popular slicers [pop22], *Ultimaker Cura*. This plugin will configure and generate *Fabric Faces* structures, by using the *Blender* plugin , that will be printable with the current slicer settings regarding build volume and nozzle size. It will also contain a generator for an interlock system, to fix the fabric in place.

A *Cura* plugin will be developed, to make *Fabric Faces* more accessible

3.1 Blender plugin extension

To let the slicer settings determine our *Fabric Faces* parameters, we first have to start with the extension of the *Blender* plugin , beginning with the introduction of a few global variables with default parameters:

```
printer_bed_width = 230
printer_bed_height = 230
printer_nozzle_size = 0.4
printer_layer_height = 0.2
```

These variable allow us to calculate the frame thickness and the connector size, so that everything can be printable.

Minimal connector size is set to be 5 time nozzle width and 5 times layer height

First, we will start with limiting the minimal size of the connectors, because they will be the smallest features generated on the frame. By comparing different widths and heights (fig. 3.1), we determined that the width should be at least 5 times the nozzle width and the height should be at least 5 times the layer height, to ensure the connectors printable while also retaining the ability to clip together.

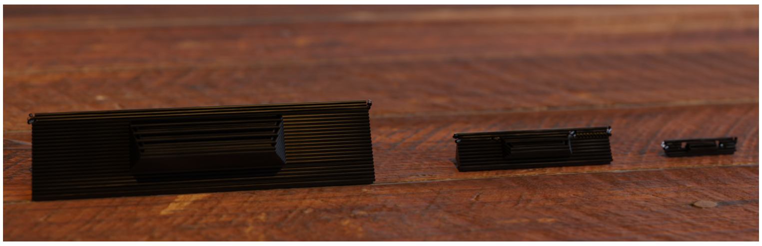

Figure 3.1: Rendered example of different sized connectors

Minimal frame width is set to be 10 times the nozzle width

With the minimal connector size determined, the minimal thickness of the frame has to be 10 times the nozzle width, so that the connectors can fully rest on the frame (fig. 3.2), and can also be fully embedded into it (fig. 3.3).

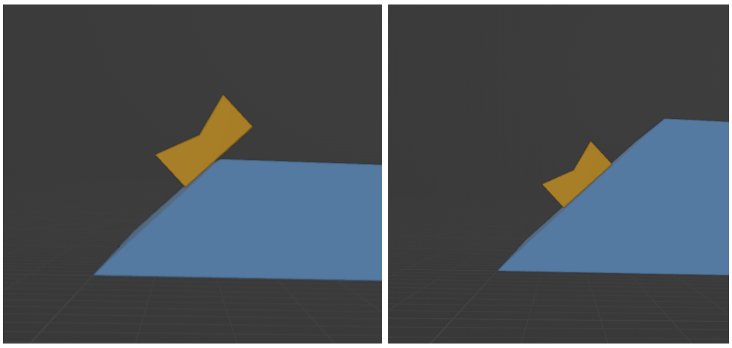

Figure 3.2: Screenshot of *Blender* with different frame widths with a connector on top

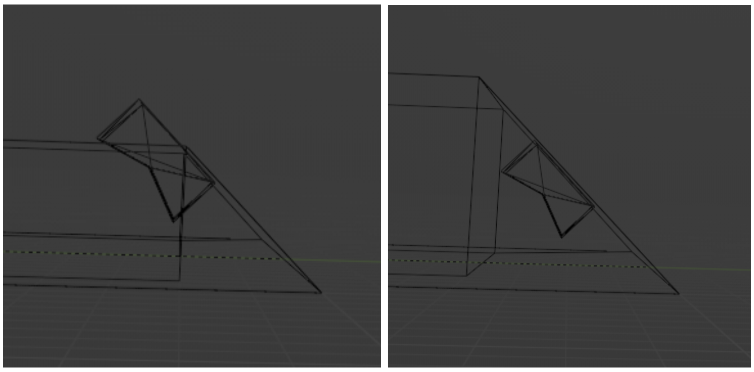

Figure 3.3: Screenshot of *Blender* with different frame widths with a connector embedded

Continuing with limiting the maximum frame size to avoid generating oversized frames (fig. 3.4). To limit the size of the frame we have to modify the already existing `grow_island` function.

GROW_ISLAND:
This function is responsible for deciding, whether or not the frame could be extended, starting with a frame of one face and checking if the growth of the frame, by including a neighbouring face, results in an intersection with itself. If no intersection is detected, the frame is extended and the procedure repeated. If an intersection is detected the frame is not extended and other possibilities are checked. If growth is no longer possible without generating intersections a new separate frame part will be generated and the growth procedure repeated until all faces are included in one of the generated frames.

Definition: *grow_island*

Furthermore, we add an additional verification to the intersection check of the `grow_island` function to determine whether or not the bounds of the extended frame will exceed the print bed. To calculate the bounding box (the smallest rectangle which completely encases the part) we take the smallest and biggest x and y coordinate of the vertices of the frame. If this bounding box exceeds the print bed size, this extension will not be allowed (fig. 3.4).

`grow_island` is extended to check if the bounding box of the frame exceeds the print bed size

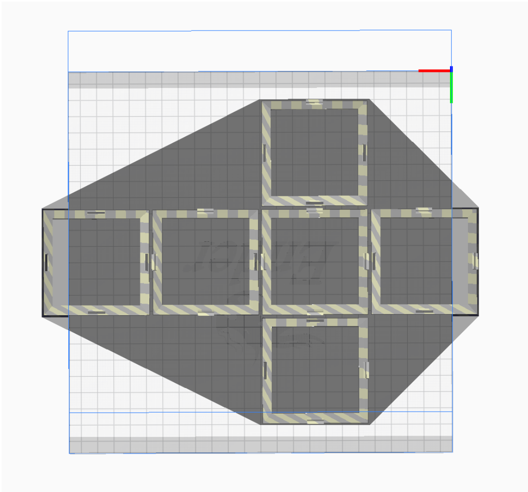

Figure 3.4: Screenshot of *Cura* with a oversized *Fabric Faces* frame.

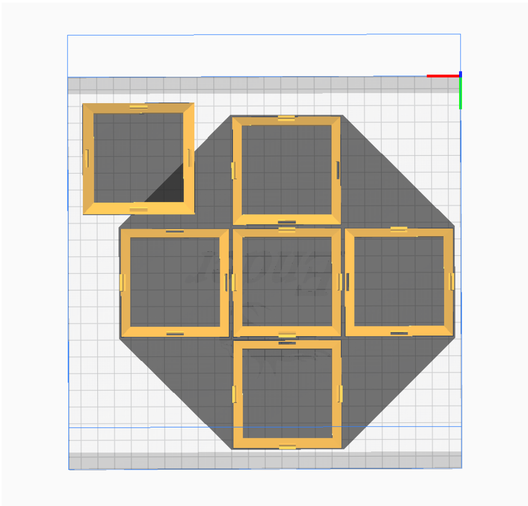

Figure 3.5: Screenshot of *Cura* with a multipart *Fabric Faces* frame generated with the implemented print bed size limit.

3.2 Cura plugin

With the *Blender* plugin able to constrain our object to our slicer settings we can start the development of the *Cura* plugin.

3.2.1 Interlock system

We first have to choose, what kind of interlock systems we would like to use. To improve the accessibility we decided to use a printed interlock system, which could be 3d printed on the print bed and hold the fabric in place. Here we test to approaches, a reusable (fig. 3.6) and a non reusable. (fig. 3.7)

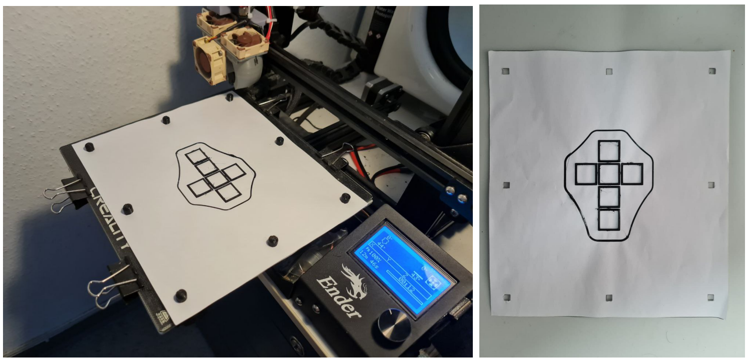

Figure 3.6: Image of the reusable interlock System

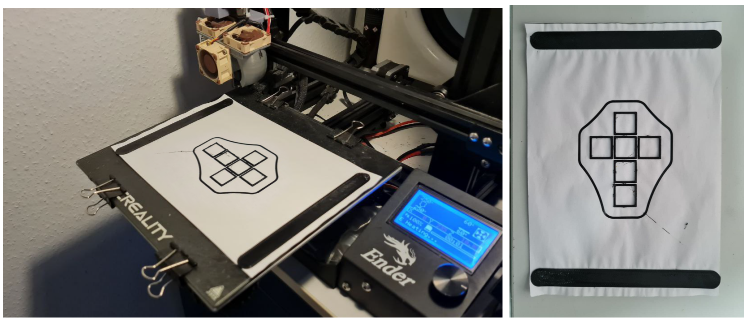

Figure 3.7: Image of the non reusable interlock System

For the reusable approach we designed a pin (fig. 3.8) which is printed around the build plate (each pin can be printed in succession with the *Cura* print setting "Print Sequence: Once at Time"). The pins can be printed in about 10 minutes with the default *Cura* print 0.2 mm settings for PLA and can fixate flexible and stiff fabrics.

Reusable interlocks: Pins are printed on the build plate, which can hold the fabric in place

The non reusable approach is a two staged print. First, two thin strips are printed in parallel on the print bed (fig. 3.9). Then the printer is paused to wait for the user to lay

Non reusable interlocks: Fabric is fused to the print bed with a strip of printed plastic

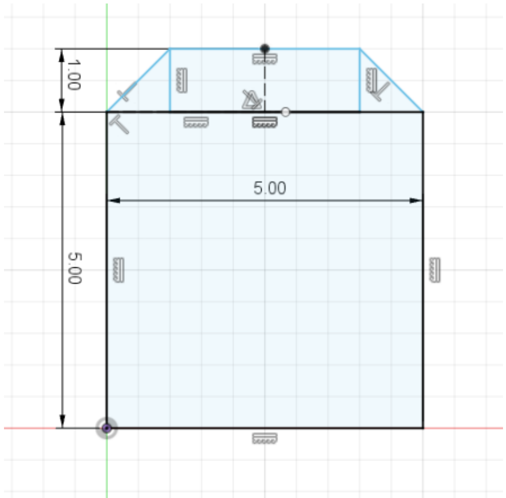

Figure 3.8: *Fusion 360* sketch of the interlock pin

down the fabric by using the gcode command `M0`. After the user confirmed to the printer, that the fabric is in place, a few layer are printed on top of the fabric and the strip (fig. 3.10) to secure them in place by fusing the fabric to the strip. Testing revealed that this approach struggled with elastic materials, which has to be hold in place while the printer fuses it to the strips, while also taking 30 minutes with default 0.2 mm settings for PLA.

Therefore, the reusable approach is chosen to be integrated in the *Cura* plugin, because of its higher reliability and faster print times.

A svg exporter for the interlocks will be created, to ease up preprocessing

To be able to use the interlocks the fabric has to be precut, which adds one additional step to the preprocessing of the print. To support the user in this step, we will integrate a function within the plugin, which creates a .svg (vector graphics) file that contains the places which have to be cut. With this .svg file a template or a stencil could be created, to aid the preprocessing step or make use of other machines to precut the fabric, like a laser cutter (if the fabric can be cut safely).

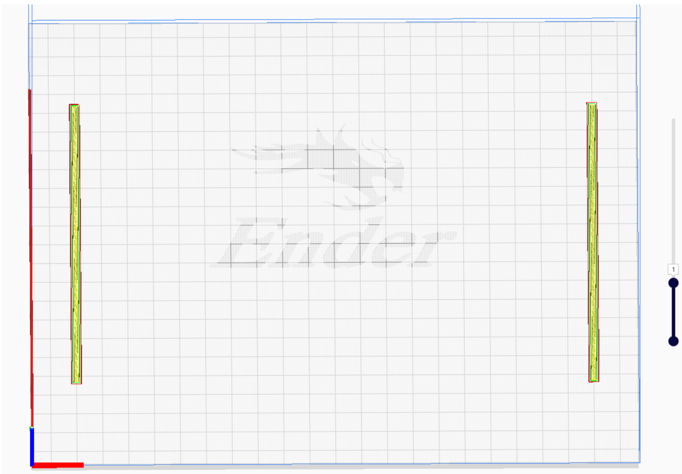

Figure 3.9: *Cura* screenshot of first layer of the non reusable interlock

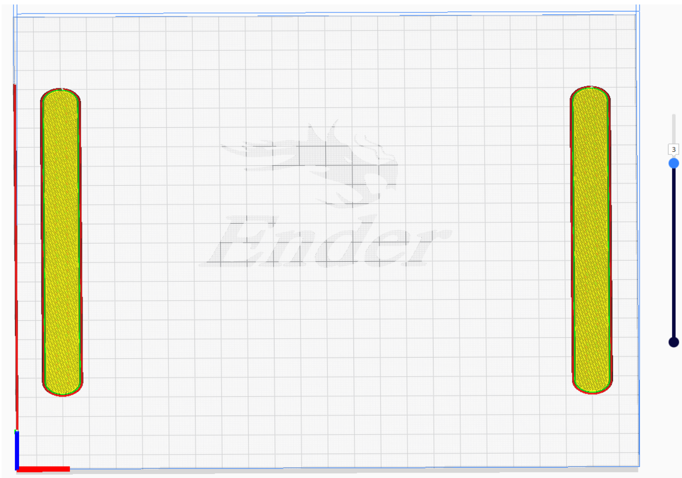

Figure 3.10: *Cura* screenshot of an upper layer of the non reusable interlock

3.2.2 Plugin development

The *Cura* plugin will provide an interface (fig. 3.11) to access the *Fabric Faces* structure generation and the creation and slicing of the interlock system with the svg exporter, to provide a vector graphics file of the positioning of the interlocks. Another feature will be a queue slicer to automate the slicing of the *Fabric Faces* structure, even if it is a multipart

An Interface will be created in *Cura* to access all the *Fabric Faces* features structure, this queue slicer will modify all generated gcode to avoid possible interlock points.

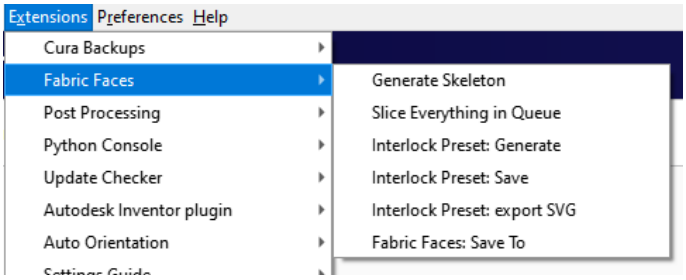

Figure 3.11: *Cura* screenshot of the *Fabric Faces* plugin menu

Generate Skeleton

Starting with the *Fabric Faces* skeleton generation (this will handle the conversion of the base part into a *Fabric Faces* structure), we first have to write a script, which will be handling all blender operations. This script will expect 7 arguments:

- `plugin_path`: path of the *Fabric Faces Blender* plugin
- `unfold_filepath`: path of the base object
- `save_filepath`: save path for the *Fabric Faces* strucure
- `pb_width`: width of the print bed
- `pb_height`: heigth of the print bed
- `p_nozzlesize`: nozzle width
- `p_layerheight`: layer height

We create a script which will be executed in blender to handle the unfolding and configuring of *Fabric Faces*

With these arguments given we begin the implementation: We start by importing and enabling the *Blender* plugin from the given `plugin_path` to make sure we can access *Fabric Faces* . Next we delete all objects which are in the scene to have a clean slate. Then we import our object from the `unfold_filepath`. We now have to rebuild our scene context:

```
new_context = bpy.context.copy()

new_context['area']=
[a for a in bpy.context.screen.areas
if a.type=="VIEW_3D"][0]
```

This step is necessary in the current *Blender* 3.0 version if *Blender* is stated in headless mode (which it will be). If we do not rebuild our context, the *Fabric Faces* algorithm crashes due to missing information on the rotating of the connectors.

HEADLESS BLENDER:
Blender can be accessed without starting the GUI, this enables the use of *Blender* through a command line. *Blender* can be also started with a python script attached, which will be executed of startup

Definition: *Headless blender*

Now that we can access *Fabric Faces* inside *Blender* , we try to unfold the object. If this succeeds we export all generated *Fabric Faces* structures to the `save_filepath` and exit blender.

Circling back to the *Cura* plugin. We now have to check, if there is a blender version installed, therefore we have to check the windows default install directory of *Blender* (windows was chosen because it is the most used desktop operating system [opS22]), and select the highest version we can find. Afterwards we check if an object is selected, if no or multiples are selected, we throw an error message suggesting to select exactly one object. If an object was selected, we then extract the printer dimensions and slicer settings (we will have to pass to the *Blender* script) to configure *Fabric Faces* . We then execute *Blender* in the background and try to generate the *Fabric Faces* structure. If the structure cannot be generated, an error message is displayed. If structures were generated successfully we then proceed to remove our object from the build plate, and import all generated structures.

The *Generate Skeleton* button will execute blender and use *Fabric Faces* to generate the skeleton and replace the base object with it

Interlock Preset

As discussed previously, the interlocks are pins will be printed on the print bed to fixate a fabric on which we can then print on. An interlock pin is a 3d model, which we created inside *Fusion 360*, to place the interlocks on the build plate we implement the *Interlock Preset Generate* function:

Interlocks are places and distributed evenly at the edge of the build plate

To place our interlocks we first move every object places on the build plate away to the side so that they would not interfere with the interlocks. Afterwards we determine our print bed size to spread out the interlocks at the edge of the build plate. For reliability reasons we choose to have a small offset from the edges of the print bed to compensate for any misalignment a printer could have. We choose 15mm as an offset. With our offset determined, we place the interlocks at the edge of the build plate, with the center of each pin 15mm away from the edge and evenly spread out (fig. 3.12).

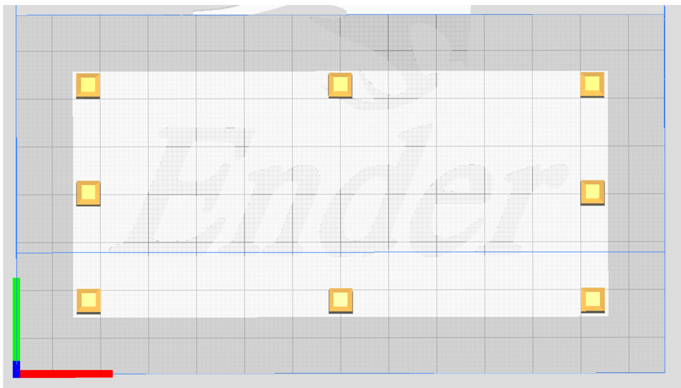

Figure 3.12: *Cura* screenshot of the *Fabric Faces* interlock pins

The .svg exporter generates a vector graphic which is a 2d representation of the interlock pins

With the placement determined we can implement the .svg export function: First, we ask for a save location and name. Then we proceed with generating the vector graphic. We will use the same placement algorithm we use in our interlock generator. With the coordinates known we can draw quads at each position of the pins, and save the .svg file at the chosen location.

Last we add the *Interlock Preset: Save* function, which slices the interlock pins, and deletes them from the build plate.

Queue Slicer

With our Interlock System, we have to avoid the pins at the start of the print to prevent knocking them over, this requires modification of the gcode file. Additionally, a functionality to slice a multipart model in succession to a save location has also to be implemented to improve the workflow if a multipart *Fabric Faces* structure has been generated. We combine these two requirements into one function: the *Queue Slicer*.

Gcode modification and queue slicing will be implemented together

We start by moving all our objects from the build plate and begin slicing each object one by one by moving them onto the build plate in succession. After an object is sliced, we modify the generated gcode, by adding a jump to the gcode. For this we first remove any steps before the first layer of the object starts (this is marked in gcode generated by *Cura* with the comment `;LAYER:0`), then we move the print head up 20mm, next move it to the center and then move it back to its initial height. After this jump the printer moves to its starting position and the print is started.

The queue slicer slices every imported object in secession, and modifies the gcode for avoidance of the interlock points

Chapter 4

Evaluation

After we integrated *Fabric Faces* into *Cura* , it is left to evaluate its performance in comparison to a standard *3D-Printing* approach. We will split this study into two parts, in the first part we will compare the print time and material use of of objects with different print settings to the unfolded *Fabric Faces* version. In the second part of this study, we will compare the impact resistance of an object, printed with different settings and printed with the *Fabric Faces* approach.

We will split the study into two parts, first the print time and material use analysis, and last the impact resistance test

4.1 Print time and material use evaluation

Starting with the print time and material use analysis, we first have to acquire a set of sample objects to conduct our evaluation on, and then create a setup for comparing and evaluating of different print settings and approaches.

4.1.1 Setup

Generation of test objects

To get a suitable set ob samples we first have to consider the limitations of *Fabric Faces* . Due to compute time *Fabric*

Due to the *Fabric Faces* limit of 100 polygons, we will create our own test sample objects

Faces is limited to 100 polygons. This limits our choices of objects, so a random set of open and freely available samples may not be unfoldable, without modification. So we choose to generate the objects ourselves, which will be created within the 100 polygon limit of *Fabric Faces* , but with enough variability to reflect a broad spectrum of print scenarios.

geometry nodes are used, to create our test samples

To generate the objects we will use the *geometry nodes* of *Blender 3.0* combined with some *modifiers* to produce a set of different object. The goal is to create an algorithm, which will create objects with different geometries to reflect different kinds of print, an some challenges like overhangs or sharp edges. Starting with the *geometry nodes*, we begin

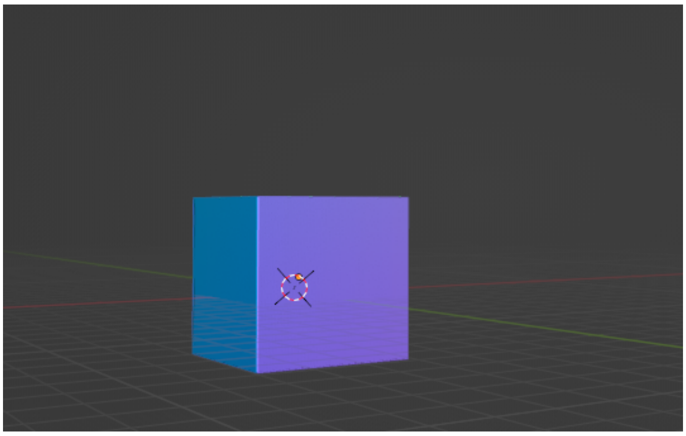

Figure 4.1: *Cura* screenshot of the *Fabric Faces* interlock pins

A cube is covered with more cubes, scaled and rotated at random

with a 20x20x20 cube (fig. 4.1), here we use *distributed points on the faces* of it to mark random spots on each face of the cube with points, the number of them is determent by a random seed number (every random number is derived by this same seed inside this generator). Then we create new cubes instances on top of each point (each cube instance (including our base cube) is randomly rotated and randomly scaled independently in x, y and z direction), the sizes of the individual cubes is driven by a 4d Noise Texture, available as a parameter inside the *geometry nodes*, which produces a satisfying variability in scale in each axis. With the cubes distributed, rotated and scaled we merge all cubes into one objects.

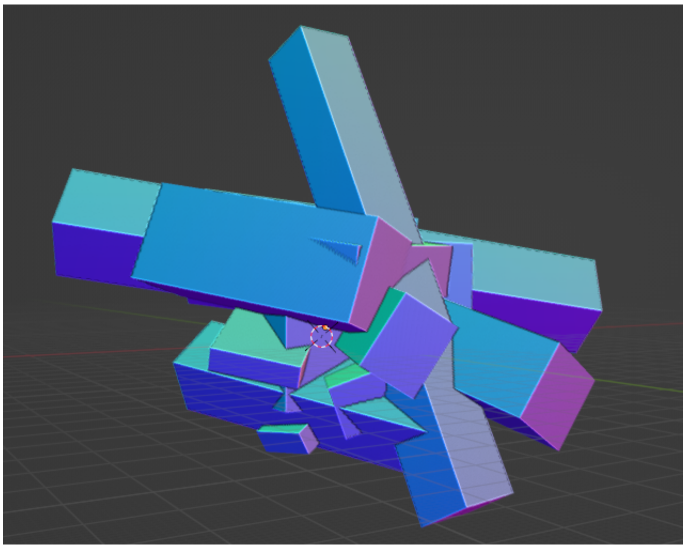

Figure 4.2: *Blender* screenshot of a cube after we apply our geometry nodes

This leaves a abstract looking object (fig. 4.2), we have to modify to make it more representative of a real part, and make it more printable. Starting with a remesh (fig. 4.3), this smooths the object to a degree, and limits our polygon count, but still generates overhangs or sharp corners we wanted to test.

The modified cube is smoothed by remeshing it

Secondly we would like to have a flat bottom to make it printable without rotating or en excessive amount of support. In most cases we would ideally have to orient every part so that the amount of support is minimal and the part has the best chance of adhesion to the print bed. To avoid this step every object will have a flat bottom face, this also benefits the standard *3D-Printing* approach, by reducing support structures, because big parts of the objects will be supported by themselves. To achieve this we first move our object from it default position (its center is located at 0.0) up by 10 (to reduce the amount of geometry which will be cut), and perform an intersection Boolean with a big cube (representing a print volume) with its bottom face at $z = 0$ (fig. 4.4), this generates objects which will be printable inside a predetermine print volume with a flat bottom face.

The modified cube gets a flat bottom to make it print easier, without further transformations

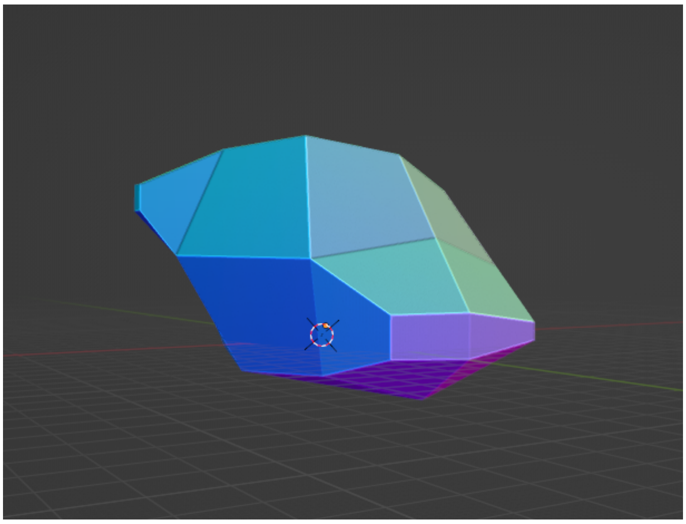

Figure 4.3: *Blender* screenshot of our modified cube after we apply a remesh

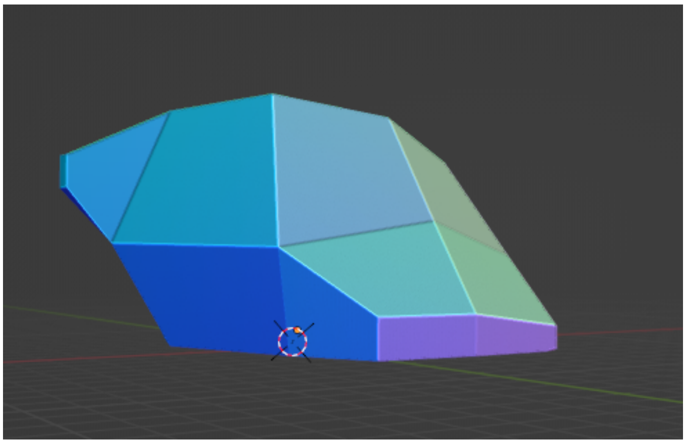

Figure 4.4: *Blender* screenshot our modified cube we cut the bottom flat

Preparation for evaluation

To evaluate the print time and material use, we first have to determine the print settings we will be comparing, for that we will use the *Cura* default settings with some modification to *wall line count* and *infill amount*. As a printer we choose the *Creality Ender 3*, as the most bought and therefor probably one of the most popular 3d printers according to Amazon [bes22].

We modify the default settings to represent fast and more time consuming prints:

- walls $= 2$, infill 20%, we call this standard
- walls $= 3$, infill 20%, we call this slicer option so_203
- walls $= 3$, infill 0%, we call this slicer option so_003
- walls $= 1$, infill 20%, we call this slicer option so_201
- walls $= 1$, infill 0% , we call this slicer option so_001

4.1.2 Procedure

100 Samples are generated

First we start to generate our samples and try to unfold them to get a sample size of 100 objects. We use headless blender with a python script, to generate each object, unfold them with the *Fabric Faces* algorithm, and export both versions of the part. After the generation, each *Fabric Faces* structure has to be checked, whether or not it generated successfully. In some cases a unfavorable object with difficult geometry will lead to an unsuccessful generation (fig. 4.5) of the *Fabric Faces* structure, where only the connectors will be exported, and the skeleton won't full generate. In our case, we had to export 364 samples, to get our desired 100 sample pieces, which translates to an success rate of $\frac{100}{364} \approx 27,5\%$ in our specific scenario, with our generation method.

Figure 4.5: An unsuccessfully generated *Fabric Faces* structure

Now that we acquired our sample set, we generate our gcode files by utilizing our queue slicer created for our *Fabric Faces Cura* Plugin. We first import all of our models, set our desired settings, and let this tool generate our files.

The time and material use estimations printed in the gcode files will be used for evaluation

For our evaluation we will use the time and material use estimation by *Cura* , which prints them inside a comment in every generated gcode file. To confirm this estimations as usable for this study, we printed the first 3 gcode files, of the standard variant, on an *Creality Ender 3* and we could confirm, that the time estimation is accurate. The Material use (given in meters) is also assumed to be correct, because the e-steps of the extruder of this particular printer, are calibrated to the material used, to extrude the exact material amount, the slicer expects. This estimations could vary by different printers, but this evaluation will be still representative, because we will evaluate the relative difference between all printed variants and not the total amount used.

The time and material use estimations are evaluated and compared to each other

After generating all the gcode files we plot every time and material use estimate into *Excel*, compare then to each other and sort them by print time of the standard setting. An additional point of comparison is derived by a feature of *Fabric Faces* , namely the possibility of parallel printing of frame structures, if some structures could not be generated in one pass, either because the geometry would not allow it, or the resulting frame would not fit on the build plate. So there will be two different comparison point of *Fabric Faces* , the first will assume, that if there are more than one generated frame part, that the parts will be printed in succession and we will call this time *Total Frame Time*, the second will assume, that every part could be printed in parallel, we will call this *Maximum Frame Time*.

4.1.3 Discussion

Beginning with the print time estimation comparison (fig. 4.6) (the graph shows the time estimated in seconds)

- the standard variant (blue)

- the so_001 variant (orange)
- the *Fabric Faces* Total Frame Time (brown)
- the *Fabric Faces* Maximum Frame Time (dark blue)

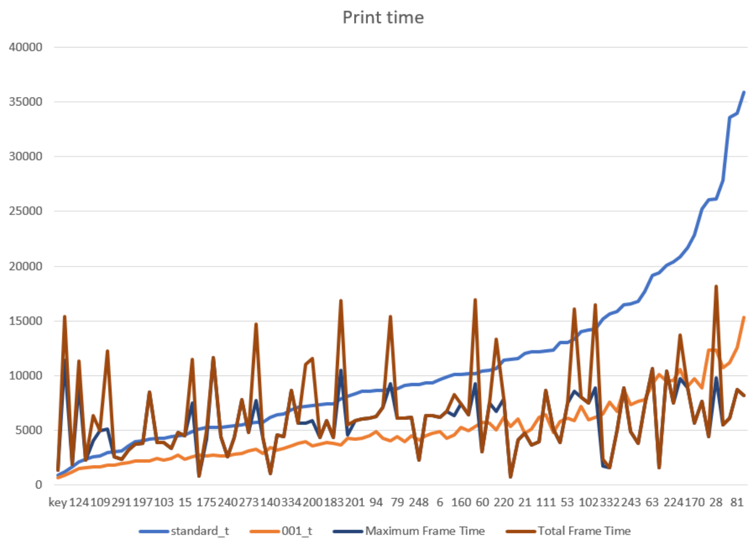

Figure 4.6: The graph of all times extracted from our gcode files

Fabric Faces is more advantageous, if the base object is larger, small prints are less efficient as *Fabric Faces* strucutre

We notice, the smaller an object is, the less advantageous *Fabric Faces* becomes, even adding print time in some cases compared to the other versions, this is due to the increasing amount of bottom layers, which are printed slower in the default *Cura* slicer options, and an increase of material used, due to a higher density of the part (fig. 4.7), determined by the minimum frame width set in the plugin. We also notice, that the so_001 is mostly as fast or even faster than *Fabric Faces* printed in parallel, but this advantage decreases with increasing part size, so that, especially for the parallel approach of *Fabric Faces* , where we see, that due to the size limitation of each frame which could fit on a print bed, this printing technique tends to scale much slower. We also see that, even though the standard variant is mostly the slowest option, *Fabric Faces* could be slower in specific scenarios, where for example, the part is long and thin, which results in more walls printed on the standard variant, which could be printed faster than the bottom layers needed for *Fabric Faces* .

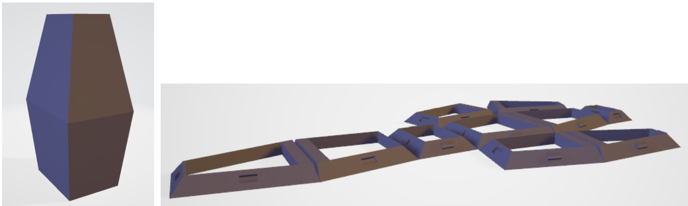

Figure 4.7: A model unfolded by *Fabric Faces*

Continuing with the material use estimation comparison (the graph shows the length of the filament estimated in meters)

- the standard variant (blue)
- the so_001 variant (orange)
- the so_003 variant (grey)
- the Fabric Faces Total Frame Time (yellow)

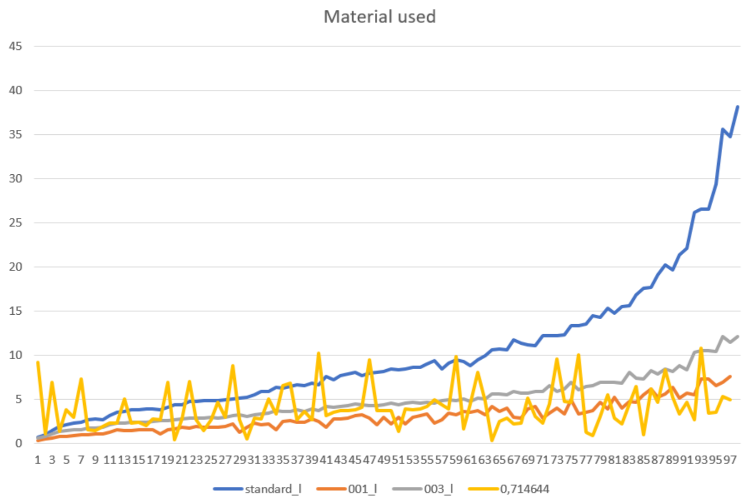

Figure 4.8: The graph of the material use estimation extracted from our gcode files

Here we see, that the so_001 is mostly still the most efficient variant, but the difference is not as pronounced compared to *Fabric Faces* as in the time estimations, we can also notice, that again, the bigger the part gets, the more advantageous *Fabric Faces* becomes.

4.2 Impact resistance evaluation

Now that we evaluated the print time and material use estimations, we will test the impact resistance of each print variant, to determine, which printing technique could be the most advantageous, in regards of part strength. We will also evaluate the difference between a frame printed on a fabric, or just left bare. To test the impact resistance we will use a variant of the *Izod impact strength test*, which is an industry standard impact resistance test [MSAA21].

Impact resistance is measured by a variation of the izod impact strength test

4.2.1 Setup

First we create our test samples, which are 20mm x 20mm x 100mm cuboids. We will print them once in every variant we decided on in our time and material use estimation, and printed the *Fabric Faces* part once on a polyester fabric, and once without a underlining fabric. As a filament we will use *Ultimaker PLA - M0751 Yellow 750* . Additionally, each object not printed as a Frame will be printed once in a horizontal (suffix ”h”), and once in a vertical position (suffix ”v”) to consider layer adhesion.

20mm x 20mm x 100mm cuboids will be tested for impact resistance

To test the impact resistance of this parts, we construct a frame out of aluminium extrusions (fig. 4.9), and fix it on a table. We connect a 1.8 meter, 40mm x40mm aluminium extrusion as our ¨hammer¨to our frame an lubricate the shaft on which its hanged on. Then We fix a wooden plate to the same table, with a rectangular hole cut in it, so that the part, we will fix in it, would be hit by the tip of our hammer arm. we then position a camera on the same height as our rotation axis of the hammer arm focused on the arm and set to record with 100FPS (frames per second), to determine the strength of the energy absorption of each part on impact by evaluating the maximum angle reached by the arm.

A frame out of aluminium extrusions is created to conduct the test

4.2.2 Procedure

Before we start printing, we first have to prepare the Fabric

Fabric is laser cut to reduce preprocessing steps

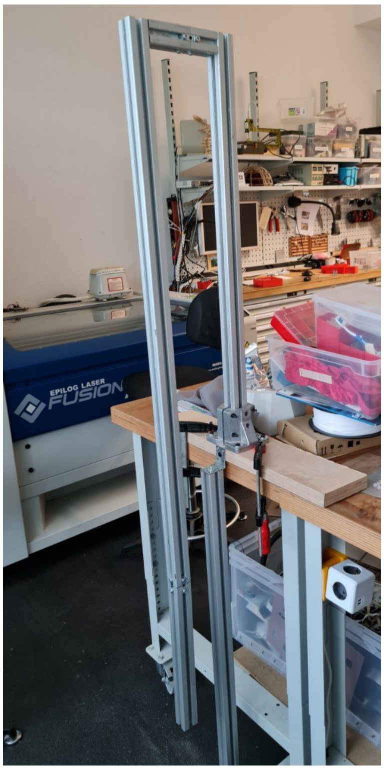

Figure 4.9: Variant of the izod impact strength built out of aluminium extrusions

our *Fabric Faces* structure could be printed on, for that we used the *Epilog Fusion M2 40* Laser Cutter , to cut out the intersection pin holes. The settings used to cut the polyester fabric where:

- Strength: 80
- Speed: 80
- Frequency: 100

With the fabric prepared, we can print out all our parts (fig. 4.11) and test the impact resistance of each part, fixating the printed parts to the aluminium structure, then lift the hammer arm (to a horizontal position), and letting it go.

We then analyze the amount of absorbed strength as a fraction of an angle the arm could reach after impacting the part

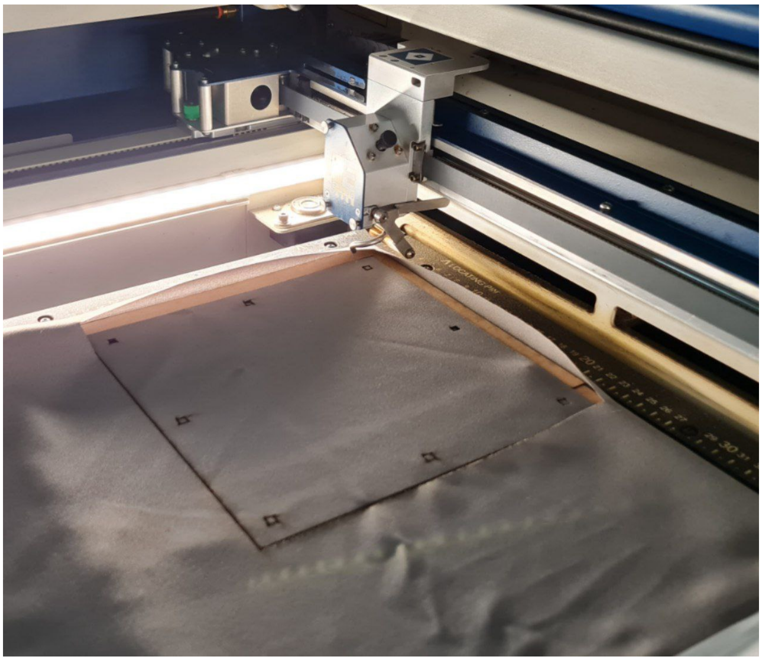

Figure 4.10: Epilog Fusion M2 40 cutting fabric for *Fabric Faces*

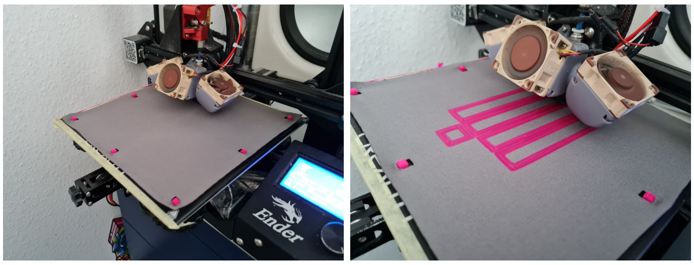

Figure 4.11: *Fabric Faces* structure printed on a laser cut polyester

(fig. 4.12). 100% means the arm could not break the part, so we measure an angle of $0°$, and 0% means, that there was no strength absorbed, meaning we measured an angle of $90°$.

Before we start testing we have to determine a baseline by measuring the friction loss by letting the arm swing without any part in it. Then we can strap in all of our parts, and begin breaking.

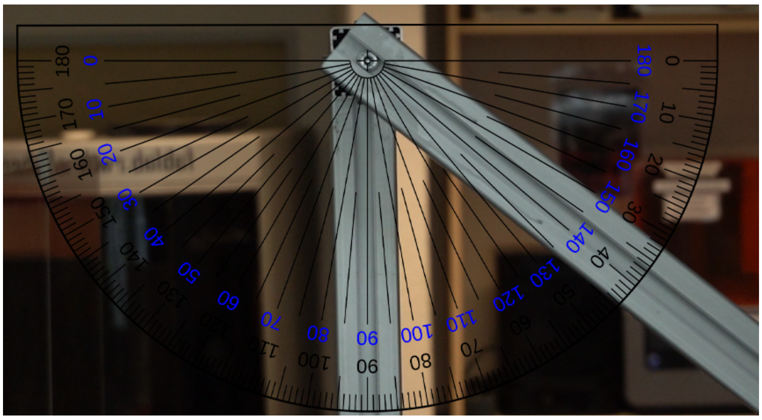

Figure 4.12: Example of the angle measurement

4.2.3 Discussion

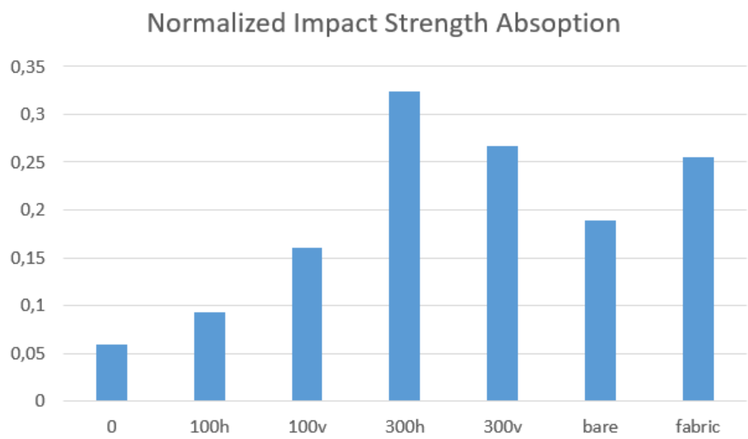

Figure 4.13: Comparison of impact resistance

Parts with infill could not be broken. *Fabric Faces* is comparable to the 300_h and 300_v versions

After conduction the tests, we notice, that our setup was unable to break any part which used the 20% infill setting, on the other variants we can see, that the *Fabric Faces* structures compared pretty favorable with the 300_h and 300_v variant (fig. 4.13), and that the 100_h variant has next to no impact resistance (0 means, that there was no part inserted, so it's basically only friction loss). Here we could also observe on another strength of Fabric Faces. Because everything is printed flat on the print bed, we don't have to worry about layer adhesion.

Fabric Faces could not broken, just bend

We then proceed to evaluate the damage on the parts themselves. While the parts printed in the classical fashion broke at roughly the point of impact, the *Fabric Faces* Variant printed without a fabric had all of its walls separated from each other because of the lack of support of an underlining fabric, to hold them together. Evaluating the *Fabric Faces* variant printed on a fabric we noticed it was just bend (fig. 4.14), and after checking the inside of the object, we could notice, that it is not broken. So while absorbing some strength while bending, the tensile strength of the fabric prevented the part from breaking.

Figure 4.14: Comparison of 100_v, 300_v, *Fabric Faces* and 120_l (top to bottom)

4.3 Result of the Study

After conducting the two parted study, we now can conclude, that *Fabric Faces* , while not the fastest or the most efficient in every case, can be useful to reduce print speeds and material use, especially with bigger parts while not compromising as much strength as you would have to, if you choose the fastest print settings discussed in this paper.

The queue slicer and laser cutter improved the workflow of *Fabric Faces*

This study also has shown, that by using the *queue slicer* to slice large amount of objects, and the .svg exporter, to preprocess the fabric by using a laser cutter , we improve our workflow by saving time and reducing the amount of work needed in preprocessing. The printed interlock pins also showed, that they where reusable and could hold the fabric in place, but they struggled with the prevention of stretching at the center, which resulted in a few failed prints.

Fabric Faces could be used for rapid prototyping, if the object is large enough

Previous papers also discussed the possibility of rapid prototyping, our data suggest if your part has to have some more resistance or is significantly big that this could be a viable approach for this use case, on the other hand, if the focus is just a prototype without any mechanical requirements, than a possible spiralized model (a setting in which a model is printed in one go with one outer wall, often used for printing vases) might be the best possible option.

Chapter 5

Summary and future work

At last, we present a summary of this thesis and show our contributions to *Fabric Faces* . Additionally we will examine ideas for future studies, which could expand on *Fabric Faces* and this thesis.

5.1 Summary and contributions

In this bachelor theses we expanded *Fabric Faces* by integrating it into *Cura* and allowing the creation of *Fabric Faces* structures inside the slicer itself. We introduced an interlock system for fabrics, which can be printed on the print bed and lock the fabric in place. We also included a .svg exporter, to create a vector graphic of the interlock points to improve the prepossessing by either including systems like laser cutting or the creation of a precise template for cutting by hand. We also modified *Fabric Faces* to be printer aware. Further we integrated a queue slicing feature, which handles necessary gcode modification for a print using the interlock system, while also providing the possibility to slice every object imported to *Cura* in succession.

We integrated *Fabric Faces* into *Cura* , and extended it with the printable interlock system and improved workflow

We also tested *Fabric Faces* mechanical and qualitative prop-

We conducted a two parted technical study erties, by comparing print time and material use to a standard approach to printing, and testing the impact strength of these printing methods.

We conclude that the workflow could be improved by including *Fabric Faces* into *Cura*
Fabric Faces can be viable for rapid prototyping in special cases

Our results showed the workflow of *Fabric Faces* could be improved ,with the inclusion of *Fabric Faces* into *Cura* and the other features shown in this thesis, especially interlock system integration. Further we showed that *Fabric Faces* can be a viable option for rapid prototyping if the model is large enough or optimized for *Fabric Faces* , but for smaller prints other methods could be more advantageous.

5.2 Future work

More mechanical properties should be tested

In future works, the over all mechanical properties of *Fabric Faces* could be examined in more detail, with comparison between different fabrics and their effect on the properties. At the moment we have one comparison point, the impact resistance, so a more detailed analysis could present interesting properties of *Fabric Faces* .

A user study could test the new workflow

Further a user study for workflow analysis of *Fabric Faces* including the *Cura* plugin and integrating other systems like laser cutter or preprinted fabric will be necessary to conclude whether or not the workflow could be improved significantly with this integration.

Another possible future development could be an extension to interlock .svg exporter, by including the first layer of the print in the vector graphic too.

Appendix A

Cura Plugin Workflow guide

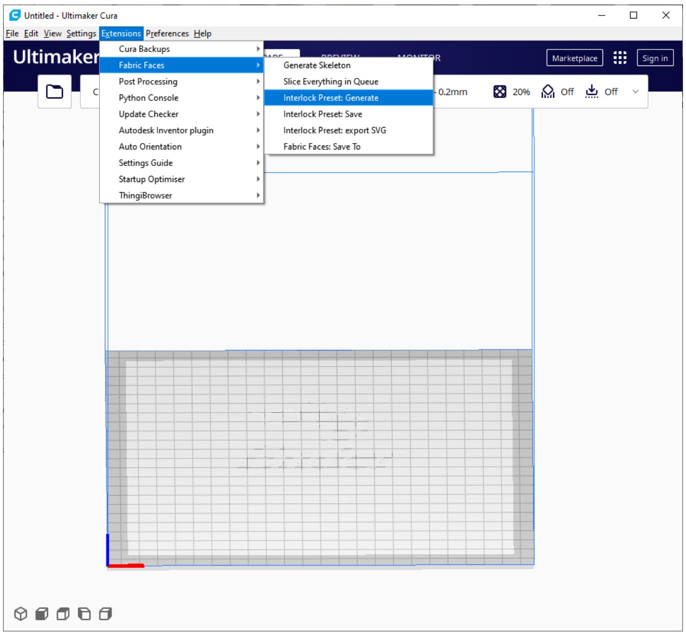

Figure A.1: First start by generating the interlock pins

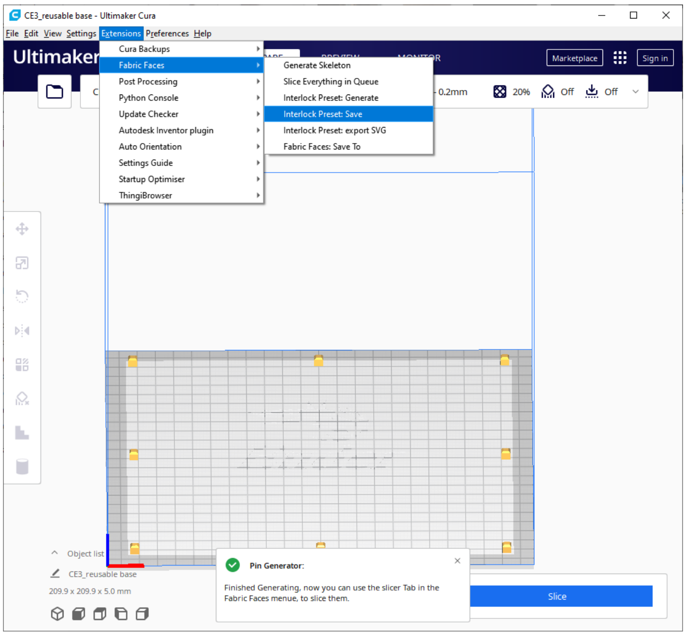

Figure A.2: Then save and print them. The pins will be automatically removed after slicing

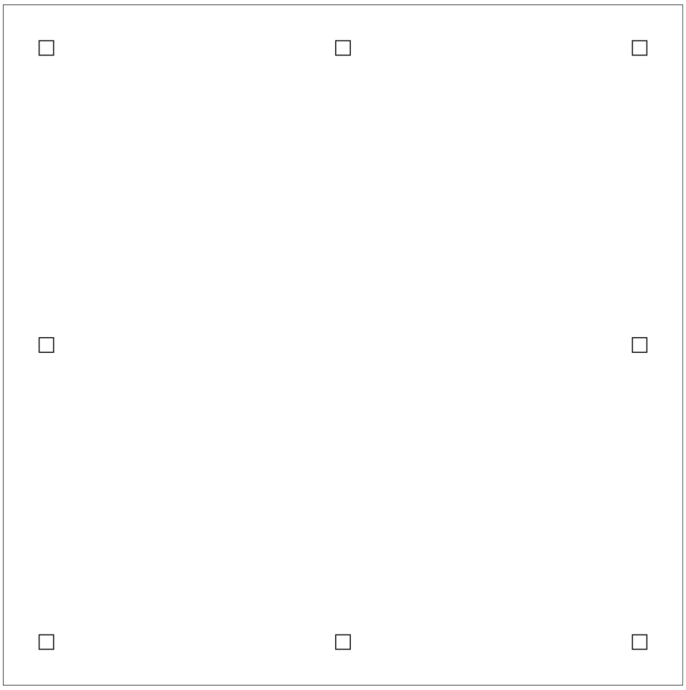

Figure A.3: A .svg file of the interlock pins can also be created

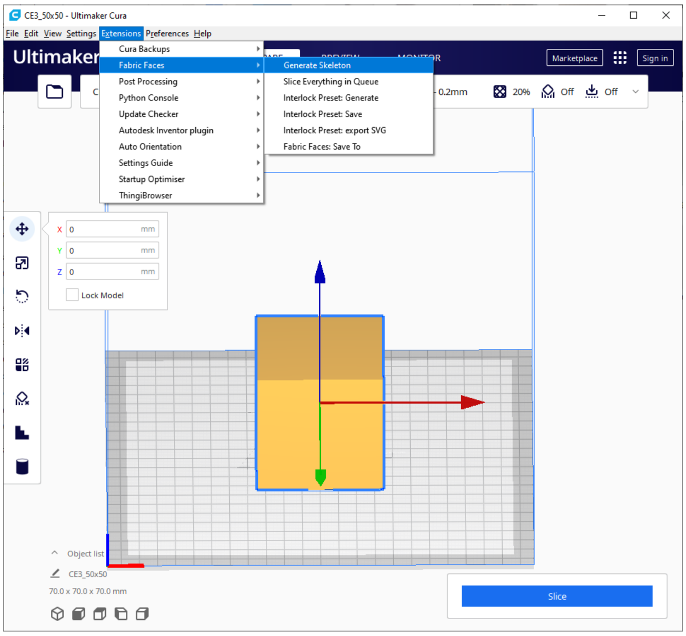

Figure A.4: Then import your object and generate the *Fabric Faces* structure

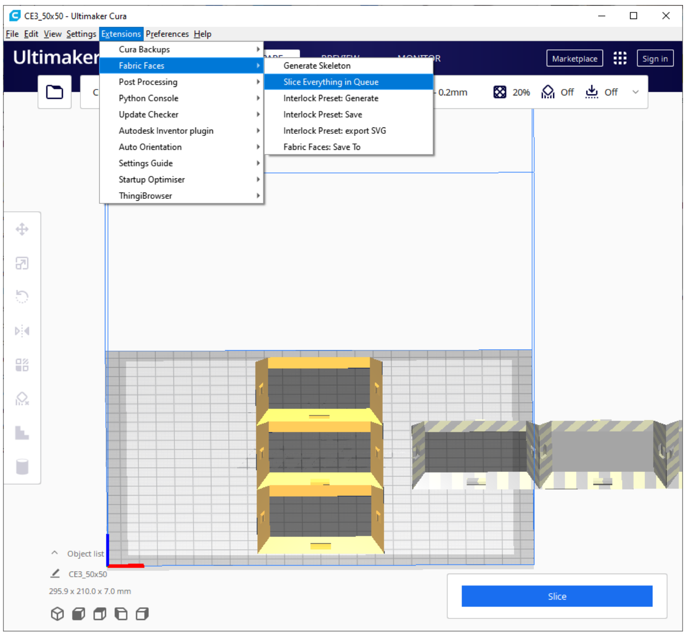

Figure A.5: At last, use the queue slicer to slice the all parts of the *Fabric Faces* structure in succession, and get the modified gcode files to avoid collision with the interlock pins

Appendix B

Technical Study Data Collection

key	standard_t	standard_l	001_t	001_l	003_t	003_l	201_t	201_l	203_t	203_l	t	l	t_1	l_2	t_3	l_4	Maximum Fr	Total Frame Time	Total Frame Length
107	918	0,667491	676	0,332364	837	0,546611	911	0,579882	975	0,750878	1349	0,714644					1349	1349	0,714644
173	1222	0,99143	920	0,444528	1082	0,76614	1208	0,860597	1397	1,11654	11407	6,75187	3975	2,39429			11407	15382	9,14616
124	1653	1,40294	1209	0,598748	1349	1,01729	1476	1,23286	1883	1,56729	1886	1,07197					1886	1886	1,07197
70	2172	1,89324	1499	0,750992	1716	1,30699	1890	1,66787	2474	2,11151	2150	1,24535	9200	5,62975			9200	11350	6,8751
189	2334	2,05456	1566	0,805805	1824	1,38808	2047	1,81897	2645	2,28274	2257	1,30005					2257	2257	1,30005
109	2591	2,31183	1704	0,855897	2003	1,53428	2269	2,0368	2960	2,57842	2258	1,35044	4068	2,40358			4068	6326	3,75402
227	2662	2,38448	1693	0,910028	2041	1,56301	2332	2,11989	3014	2,64148	4952	2,92884					4952	4952	2,92884
1	2950	2,66847	1823	0,971471	2229	1,71445	2579	2,36795	3355	2,96049	2411	1,44935	4733	2,7461	5103	3,06715	5103	12247	7,2626
291	3041	2,76601	1831	1,06165	2291	1,75625	2700	2,48528	3412	3,03992	2574	1,52276					2574	2574	1,52276
223	3152	2,62954	1945	1,06696	2537	1,8354	2762	2,31857	3551	2,93013	2358	1,39413					2358	2358	1,39413
100	3601	3,10991	2041	1,25376	2871	2,04494	3214	2,78995	4030	3,42181	3170	1,87958					3170	3170	1,87958
197	3986	3,53751	2233	1,47546	3054	2,23666	3602	3,23017	4374	3,83868	3732	2,23189					3732	3732	2,23189
360	4053	3,60791	2244	1,47399	3097	2,29134	3673	3,27993	4490	3,92992	3850	2,29607					3850	3850	2,29607
67	4232	3,77526	2233	1,45523	3168	2,34962	3782	3,41143	4690	4,12747	8474	5,05454					8474	8474	5,05454
103	4314	3,77846	2439	1,54536	3358	2,36918	3903	3,44539	4755	4,10326	3874	2,29437					3874	3874	2,29437
311	4318	3,84235	2320	1,54733	3271	2,40828	3892	3,49521	4854	4,18408	3890	2,33323					3890	3890	2,33323
57	4468	3,89546	2434	1,4816	3444	2,42807	4002	3,5109	5042	4,26846	3383	2,00988					3383	3383	2,00988
15	4545	3,80241	2769	1,02825	3540	2,53246	3940	3,19075	5328	4,39028	4808	2,74497					4808	4808	2,74497
146	4581	4,10118	2367	1,55261	3444	2,53526	4093	3,705	5195	4,48885	4494	2,62816					4494	4494	2,62816
301	4878	4,3308	2572	1,63955	3609	2,60961	4331	3,93975	5390	4,71393	4045	2,43382	7469	4,46955			7469	11514	6,90337
175	5095	4,37725	2764	1,80624	3909	2,75731	4555	3,99196	5600	4,75073	825	0,377269					825	825	0,377269
168	5252	4,69606	2633	1,70887	3788	2,79812	4726	4,25603	5810	5,12509	771	0,337138	4185	2,42138			4185	4956	2,758518
32	5263	4,71342	2772	1,87267	3829	2,79675	4806	4,34082	5752	5,07931	11625	7,02211					11625	11625	7,02211
240	5285	4,80466	2683	1,79951	3776	2,79813	4786	4,40203	5822	5,19912	4455	2,65483					4455	4455	2,65483
52	5369	4,81113	2670	1,78898	3901	2,90119	4819	4,36408	5966	5,2494	2566	1,44011					2566	2566	1,44011
105	5437	4,86608	2828	1,82695	3947	2,88896	4926	4,43864	6003	5,28621	4345	2,62674					4345	4345	2,62674
273	5506	4,96047	2927	1,92695	3994	2,90903	5046	4,56604	6027	5,35038	7812	4,65528					7812	7812	4,65528
348	5644	4,97281	3133	2,18666	4262	3,13069	5164	4,59443	6146	5,34952	4789	2,92158					4789	4789	2,92158
330	5718	5,11912	3293	1,27995	4160	3,18902	4926	4,34435	6705	5,86817	7703	4,65203	3736	2,17777	3282	1,97198	7703	14721	8,80178
140	5759	5,2325	2904	1,84361	4117	3,02058	5188	4,75917	6417	5,6985	4363	2,65469					4363	4363	2,65469
221	6228	5,44773	3417	2,24655	4600	3,25397	5742	5,04382	6752	5,84546	1043	0,514702					1043	1043	0,514702
297	6413	5,87913	3209	2,08343	4538	3,31236	5801	5,37858	7049	6,36299	4607	2,82136					4607	4607	2,82136
334	6522	5,89351	3369	2,22805	4731	3,40213	5939	5,42175	7137	6,35757	4472	2,74379					4472	4472	2,74379
364	6916	6,30427	3584	1,54366	4895	3,70919	6011	5,42661	8015	7,1539	8683	4,99786					8683	8683	4,99786
290	7090	6,26451	3789	2,42101	5102	3,57258	6519	5,80004	7693	6,72292	5678	3,32931					5678	5678	3,32931
200	7198	6,42713	3956	2,57898	5191	3,64147	6635	5,99298	7707	6,84309	5362	3,16959	5635	3,37626			5635	10997	6,54585
176	7244	6,58625	3604	2,40234	5244	3,7805	6570	6,03218	7998	7,1334	5652	3,28278	5886	3,53768			5886	11538	6,82046
270	7317	6,56838	3776	2,39797	5191	3,63384	6713	6,06837	7926	7,05175	4342	2,62826					4342	4342	2,62826
183	7392	6,81623	3921	2,74125	5335	3,91229	6812	6,34576	8009	7,28074	5926	3,61479					5926	5926	3,61479
40	7404	6,65059	3821	2,49059	5264	3,72355	6795	6,15625	8034	7,13714	4373	2,65016					4373	4373	2,65016
317	7877	7,55223	3682	1,75969	5253	4,12846	6720	6,59414	9245	8,4856	10483	6,21052	6414	3,9638			10483	16897	10,17432
201	8112	7,15452	4296	2,73242	5938	4,0851	7452	6,60864	8844	7,6892	924	0,440391	4561	2,70666			4561	5485	3,147051
294	8338	7,64092	4201	2,77227	5812	4,12353	7681	7,09991	9035	8,17538	5909	3,54662					5909	5909	3,54662
3	8550	7,80195	4268	2,80739	6027	4,21212	7856	7,2361	9343	8,35757	6059	3,64569					6059	6059	3,64569
94	8591	7,99102	4504	3,11603	6165	4,42532	7951	7,46521	9373	8,51017	6130	3,66758					6130	6130	3,66758
95	8615	7,69458	4920	3,2446	6347	4,38776	8070	7,22987	9213	8,14523	6267	3,82705					6267	6267	3,82705
97	8675	7,98378	4256	2,79932	6096	4,26429	7972	7,39261	9532	8,5649	7094	4,11585					7094	7094	4,11585
79	8788	8,04691	4041	2,06033	5607	4,25285	7734	7,09317	9543	8,84365	9286	5,76984	6137	3,65149			9286	15423	9,42133
195	8805	8,09644	4429	2,94949	6170	4,32866	8138	7,54152	9623	8,64449	6133	3,69838					6133	6133	3,69838
187	9123	8,3983	3992	2,18638	6253	4,52091	8002	7,45404	10320	9,31982	6144	3,69789					6144	6144	3,69789

Figure B.1: Time and material use estimation a.

key	standard_t	standard_l	001_t	001_l	003_t	003_l	201_t	201_l	203_t	203_l	t	l	t_1	l_2	t_3	l_4	Maximum Fram	Total Frame Time	Total Frame Length
248	9180	8,33933	4547	2,90612	6392	4,39933	8355	7,73968	9954	8,92875	6204	3,70878					6204	6204	3,70878
218	9194	8,37823	4143	2,16111	6419	4,53228	8056	7,41856	10392	9,31011	2281	1,34318					2281	2281	1,34318
184	9308	8,56954	4507	2,96212	6567	4,60296	8440	7,91172	10194	9,21922	6331	3,90452					6331	6331	3,90452
6	9343	8,60404	4706	3,12538	6428	4,51919	8588	8,04155	10056	9,15483	6371	3,8293					6371	6371	3,8293
323	9633	8,97465	4915	3,26912	6655	4,6531	8945	8,41829	10364	9,52621	6229	3,83742					6229	6229	3,83742
217	9899	9,31438	4303	2,29364	5961	4,57028	8683	8,30283	10596	10,1167	6766	4,12212					6766	6766	4,12212
160	10093	8,45886	4624	2,61327	7206	4,79012	9019	7,56968	11137	9,3128	1924	1,04445	6377	3,87122			6377	8301	4,91567
245	10120	9,0913	5286	3,41998	7130	4,89434	9413	8,49709	10878	9,67566	7440	4,35669					7440	7440	4,35669
78	10203	9,46088	4936	3,19511	6926	4,80418	9438	8,81123	11015	10,0939	6408	3,83445					6408	6408	3,83445
60	10212	9,29653	5376	3,57806	7165	5,00183	9525	8,72397	10952	9,86021	9239	5,37239	3863	2,16897	3869	2,3098	9239	16971	9,85116
204	10407	8,81452	5773	3,52595	7219	4,75896	9746	8,31124	10957	9,29282	3066	1,64303					3066	3066	1,64303
247	10528	9,41191	5644	3,64777	7411	5,06603	9794	8,84006	11241	9,97232	7468	4,40205					7468	7468	4,40205
220	10649	9,91025	5012	3,24023	7132	4,99626	9822	9,19316	11510	10,5955	6565	3,97638	6738	4,0923			6738	13303	8,06868
274	11447	10,6254	6235	4,16181	7987	5,60933	10707	10,0395	12195	11,1999	7900	4,73019					7900	7900	4,73019
309	11509	10,6696	5340	3,60162	8051	5,58469	10482	9,86964	12617	11,4532	720	0,302889					720	720	0,302889
21	11539	10,5818	6054	3,93035	8039	5,52708	10778	9,94739	12417	11,2189	4110	2,44446					4110	4110	2,44446
325	12010	11,7013	4734	2,89626	7837	5,84264	10674	10,509	13653	12,8688	4761	2,88184					4761	4761	2,88184
59	12154	11,368	5113	2,87901	7645	5,69358	10831	10,1796	13226	12,4283	3642	2,21721					3642	3642	2,21721
111	12206	11,1346	6161	3,91619	8345	5,71028	11313	10,4085	13093	11,8396	3963	2,32218					3963	3963	2,32218
271	12278	11,0867	6427	4,15164	8501	5,83055	11480	10,4146	13125	11,7546	8665	5,08922					8665	8665	5,08922
249	12310	12,155	4838	2,8197	7841	5,88608	10836	10,9176	14064	13,3687	5162	3,0095					5162	5162	3,0095
53	13007	12,1872	5824	3,42257	9027	6,4799	11560	10,9533	14710	13,3928	3861	2,30953					3861	3861	2,30953
328	13044	12,215	6104	3,99413	8219	5,88027	12042	11,38	13685	12,8704	7498	4,4217					7498	7498	4,4217
312	13322	12,281	5903	3,27257	8900	6,10172	11973	11,1138	14770	13,3782	8611	5,06346	1450	0,820576	6056	3,64711	8611	16117	9,531146
102	13986	13,3241	7185	4,91648	9576	6,90452	13048	12,5282	15070	14,1109	8010	4,74334					8010	8010	4,74334
263	14140	13,3288	5970	3,33771	8289	6,09501	12623	12,0977	14903	14,2959	7517	4,63526					7517	7517	4,63526
88	14269	13,4638	6208	3,53145	8911	6,41822	12743	12,2074	15322	14,5109	7580	4,62147	8908	5,37576			8908	16488	9,99723
332	15175	14,4761	6499	3,71776	8921	6,4794	13750	13,2404	15889	15,4366	624	0,273019	1722	0,971509			1722	2346	1,244528
169	15628	14,2439	7612	4,65207	10383	6,93706	14510	13,3241	16698	15,1472	1611	0,882808					1611	1611	0,882808
352	15853	15,2746	6737	3,8885	9702	6,92132	14371	13,969	16967	16,3856	5038	3,0017					5038	5038	3,0017
243	16509	14,7	8735	5,20945	10355	6,89252	15583	13,9113	16832	15,2444	8852	5,49369					8852	8852	5,49369
163	16582	15,5307	7333	3,98596	9050	6,82163	15038	14,1976	16679	16,4346	4868	2,88011					4868	4868	2,88011
320	16772	15,5397	7621	4,71293	11285	8,04887	15283	14,1939	18515	16,8567	3828	2,14736					3828	3828	2,14736
63	17734	16,8042	7780	4,61579	10569	7,40195	16702	15,8017	18878	17,9855	7383	4,42028					7383	7383	4,42028
82	19175	17,5508	9274	5,44677	10334	7,32209	17837	16,4925	18597	17,9657	10655	6,40983					10655	10655	6,40983
2	19390	17,7096	10118	6,13343	12435	8,21674	18348	16,8189	20218	18,4665	1587	0,924217					1587	1587	0,924217
224	20058	19,1171	9524	5,23752	10888	7,86646	18973	17,9657	20160	20,0379	10396	6,15634					10396	10396	6,15634
316	20357	20,2574	9569	5,57123	12050	8,39486	20450	19,3766	21966	21,5836	7480	4,67901					7480	7480	4,67901
159	20842	19,6134	10533	6,35501	11809	8,17277	20538	18,803	20729	20,2171	3959	2,37952	9762	5,91297			9762	13721	8,29249
170	21688	21,3681	9031	5,14472	12316	8,8317	19837	19,7794	22975	22,7209	8878	5,20793					8878	8878	5,20793
43	22839	22,1084	9719	5,65996	11464	8,35121	21067	20,6778	22208	22,7846	5643	3,35585					5643	5643	3,35585
180	25228	26,1587	8865	5,47581	13761	10,2948	22875	24,1529	27276	27,9931	7651	4,66856					7651	7651	4,66856
28	26056	26,5146	12367	7,30215	15224	10,4626	26487	25,6841	28185	28,1333	4432	2,65238					4432	4432	2,65238
7	26132	26,5146	12328	7,3022	15318	10,4627	26459	25,6841	28227	28,1333	9775	6,03114	3373	1,87903	5052	2,90157	9775	18200	10,81174
314	27836	29,3933	10688	6,52193	14294	10,4199	27131	28,0147	29570	31,0642	5499	3,3975					5499	5499	3,3975
81	33613	35,6204	11189	6,87776	15889	12,0992	30957	33,2731	34787	37,4278	6123	3,46224					6123	6123	3,46224
89	33940	34,7565	12593	7,58169	14725	11,4687	31968	32,8034	32691	35,8252	8705	5,3333					8705	8705	5,3333
73	35870	38,1352	15315	8,94067	15644	12,1149	36658	36,8975	35366	39,3544	8201	4,88414					8201	8201	4,88414

Figure B.2: Time and material use estimation b.

type	Normalized	mid	degrees		
			first	second	third
0	0,0592593	5,3333333	5,5	5,5	5
100h	0,0925926	8,3333333	8	9	8
100v	0,1611111	14,5	14	15,5	14
300h	0,2222222	20	-	-	20
300v	0,2666667	24	24	-	-
bare	0,2222222	20	-	20	-
fabric	0,2555556	23	23	-	-
- : no usefull data could be extracted from video					

Figure B.3: Impact resitance test data

Bibliography

[bes22] Amazon best sellers 3d printers. https://www.amazon.com/Best-Sellers-3D-Printers/zgbs/industrial/6066127011, May 2022.

[BGM+15] Dustin Beyer, Serafima Gurevich, Stefanie Mueller, Hsiang-Ting Chen, and Patrick Baudisch. Platener: Low-fidelity fabrication of 3d objects by substituting 3d print with laser-cut plates. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*, pages 1799–1806, 2015.

[BSK+19] Patrick Baudisch, Arthur Silber, Yannis Kommana, Milan Gruner, Ludwig Wall, Kevin Reuss, Lukas Heilman, Robert Kovacs, Daniel Rechlitz, and Thijs Roumen. Kyub: A 3d editor for modeling sturdy laser-cut objects. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*, pages 1–12, 2019.

[CUBTO+19] Enrique Cuan-Urquizo, Eduardo Barocio, Viridiana Tejada-Ortigoza, R Byron Pipes, Ciro A Rodriguez, and Armando Roman-Flores. Characterization of the mechanical properties of fff structures and materials: A review on the experimental, computational and theoretical approaches. *Materials*, 12(6):895, 2019.

[IG15] Ian Gibson Ian Gibson. Additive manufacturing technologies 3d printing, rapid prototyping, and direct digital manufacturing, 2015.

[MSAA21] Pradeep Kumar Mishra, P Senthil, S Adarsh, and MS Anoop. An investigation to study the combined effect of different infill pattern and infill density on the impact strength of 3d printed polylactic acid parts. *Composites Communications*, 24:100605, 2021.

[opS22] Desktop operating system market share worldwide. `https://gs.statcounter.com/os-market-share/desktop/worldwide`, May 2022.

[pop22] The top 10 best 3d slicers 2022. `https://www.3dsourced.com/3d-software/best-3d-slicer-printer-software/#:~:text=Ultimaker`, May 2022.

[RH19] Michael L Rivera and Scott E Hudson. Desktop electrospinning: A single extruder 3d printer for producing rigid plastic and electrospun textiles. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*, pages 1–12, 2019.

Index

Typeset May 16, 2023