

3D PRINTING SPEED OPTIMIZATION

BY MINIMIZING VOID PATHS

A Dissertation

Submitted to the Faculty

of

Purdue University

by

Jorge A. Garcia Galicia

In Partial Fulfillment of the

Requirements for the Degree

of

Doctor of Philosophy

August 2017

Purdue University

West Lafayette, Indiana

**THE PURDUE UNIVERSITY GRADUATE SCHOOL
STATEMENT OF COMMITTEE APPROVAL**

Dr. Bedrich Benes, Chair

Department of Computer Graphics Technology

Dr. Tim McGraw

Department of Computer Graphics Technology

Dr. Jose Garcia Bravo

Department of Engineering Technology

Dr. Juraj Vanek

Arevo Inc

Approved by:

Dr Kathrynne Newton

Associate Dean for Graduate Programs

Dedicated to Onatta, for sharing her life with me.

ACKNOWLEDGMENTS

I would like to express my gratitude to the people and institutions that have supported me during my studies.

First, to my adviser and mentor Dr. Bedrich Benes, for his guidance and support as well as for providing me with the life-changing opportunity to study at Purdue.

To the members of my thesis committee, Dr. Tim McGraw, Dr. Jose Garcia Bravo, and Dr. Juraj Vanek, for their comments and advice during the development of this research, and during my work at Purdue in general.

To CONACYT for offering its support through the international scholarship program (CVU 266004) and to Purdue Polytechnic for the Summer Research Grant.

To the CGT department administrators for their guidance and patience.

To Victor, Dave, Bedrich, and Tim for letting me be their teaching assistant.

I am also grateful to the HPCG lab and its former and current members (Michel, Marek, Alejandro, Juraj, Innfarn, Vojtech, Kaimo, Suren, Illia, Sonali, Ed, Jay, and Hao) for establishing a work environment in which I could grow as a scholar.

To Esteban for his advice and friendship.

Last but not least, I would like to thank my family. To my father, my mother (rest in peace), and my brothers for being there always.

Finally, thanks to my wife and soulmate Onatta for her support, courage, and love.

PREFACE

Recent years have seen additive manufacturing, also known as 3D printing, migrate from the realm of niche scientific study to a place of wide renown. Today, plastic-based 3D printers are available even for home users. However, as these printers' hardware has improved, the need for software to catch up has become increasingly evident.

Creating software to meet the printing needs of mainstream consumers will require a breadth of knowledge that not only encompasses computer science, but also spans across the disciplines of mechanical engineering, computational geometry, signal processing, and geometric processing (to name only a few). Similarly, work in the discipline of computer graphics requires knowledge from several fields, most of which overlap with additive manufacturing. Recent computer graphics research has thus caused ripples for additive manufacturing and for digital manufacturing in general, from the design of better CAD programs to the adaptation of several techniques to solve open problems in manufacturing.

Reduction of printing time remains an important goal for additive manufacturing researchers. Various software-centered solutions have already been proposed or implemented. These include improved toolpath generation strategies, segmentation of the printed piece, more efficient packing strategies, and improved support structure generation. Changing the piece's orientation with respect to the building plate can also affect print time significantly. However, finding an optimal orientation has proven to be a complex problem, as this orientation can depend on multiple variables (for instance, the number of layers and the amount of required support).

Our main contribution is to add a new criterion to this optimization by considering the voids in the path traveled by the printing head while creating a

layer. On each layer, the speed of fabrication is reduced by minimizing the number of connected components and the average distance between these regions. Once this new criterion is added to the optimization process, our experiments demonstrate that we can reduce the printing time by up to 45% compared to the past method for finding optimal orientations.

TABLE OF CONTENTS

	Page
LIST OF TABLES	ix
LIST OF FIGURES	x
SYMBOLS	xi
ABBREVIATIONS	xii
ABSTRACT	xiii
CHAPTER 1. INTRODUCTION	1
1.1 Background	2
1.1.1 Selective deposition	2
1.1.2 Selective binding	3
1.1.3 Selective solidification	4
1.2 Common Problems in additive manufacturing	5
1.3 Statement of the problem	7
1.4 Significance	7
1.5 Statement of purpose	8
1.6 Research Question	8
1.7 Assumptions	9
1.8 Limitations	9
1.9 Delimitations	10
1.10 Definitions	10
1.11 Summary	11
CHAPTER 2. REVIEW OF RELEVANT LITERATURE	12
2.1 Structural properties	12
2.2 Supports generation	15
2.3 Improving the building process	17
2.4 Summary	20
CHAPTER 3. METHODOLOGY	22
3.1 Overview	23
3.2 Optimization problem	24
3.2.1 Sampling the solution space	25
3.2.2 Representative slices	27
3.3 Score Function	28
3.3.1 Orientation score	29

	Page
3.3.2 Slice score	29
3.4 Determining the corresponding weights	31
3.5 Summary	33
CHAPTER 4. EVALUATION	34
4.1 Outliers	35
4.2 Summary	37
CHAPTER 5. CONCLUSIONS	39
5.1 Future work	40
LIST OF REFERENCES	41
VITA	45
PUBLICATIONS	47

LIST OF TABLES

Table	Page
4.1 Outliers of the comparison experiment	38

LIST OF FIGURES

Figure	Page
1.1 Selective deposition printer	3
1.2 Selective binding printer	4
1.3 Selective solidification printer	5
2.1 Stress analysis by FEM	14
2.2 Structurally sound hulls by pattern textures	14
2.3 Segmenting and packing a model before printing	15
2.4 Pyramidal decomposition to avoid the need of supports	16
2.5 Comparison between supports structures	17
2.6 Inner supports structures for structural soundness	18
2.7 Adaptive slicing height	19
2.8 Fermat spirals as toolpaths	20
3.1 System pipeline	23
3.2 Slicing of a model	25
3.3 Sphere subdivision for sampling	26
3.4 Representative slices	27
3.5 Trapped volume	30
3.6 Positive and negative regions of a slice	31
3.7 Objects used to estimate the weights	32
4.1 Histogram that summarizes the evaluation results	35
4.2 An example where D is more influential than H and C	36

SYMBOLS

(α_x, α_z)	Orientation of the piece with respect to the building plate
H	Height of the mesh in a given orientation
V	Trapped volume in a given orientation
C	Number of positive regions on a given slice
D	Average distance between all the positive regions on a given slice
δ_y	Layer height
r	Representative slice
$h(x)$	Projection of $x \in \mathbb{R}^3$ onto the building direction

ABBREVIATIONS

3DP	3D printing
ABS	Acrylonitrile Butadine Styrene
CAD	Computer Aided Design
CG	Computer Graphics
CNC	Computer Numerical Control
FDM	Fused Deposition Modeling
FEM	Finite Element Method
GPU	Graphics Processing Unit
PLA	Polylactic Acid
SLA	Stereolithography

ABSTRACT

Garcia Galicia, Jorge A. Ph.D., Purdue University, August 2017. 3D Printing Speed Optimization By Minimizing Void Paths . Major Professor: Bedrich Bennes.

Though the field of 3D printing has seen substantial innovation since its inception, printing speeds remain slow. The spatial orientation of a working piece on the building plate plate is a significant contributor to its printing time. We present a novel solution based on the observation that the printing head must start and stop deposition when traveling between separate regions of a printed layer. Our method finds an orientation for the object that minimizes the distance between the disconnected components, thereby minimizing the time needed for the printer head to traverse empty areas. The method also considers the height of the printed object, its trapped volume, and the number of connected components in each layer. We provide an optimization strategy that considers all four criteria, each weighted according to printer-specific and experimentally-obtained parameters. Printing speeds in our trial are improved by up to 45% by adding the criterion of disconnected component distance to the optimization.

CHAPTER 1. INTRODUCTION

The first modern numerically controlled 3D printer was created by Charles W. Hull during the late 1980's. However, it was not until 2007, with the advent of crowdsourcing and the so called *RepRap* movement, that 3D printing (3DP) became mainstream. Today, the plastic-based materials and 3DP kits are cheap and easy enough to assemble that casual hobbyists may use them.

Even users who cannot individually afford 3D printers now have access to 3DP. Public facilities like libraries increasingly host 3DP. Additionally, services that allow users to upload a model and have the corresponding working pieces delivered to their door, and even online services that find the closest available 3D printer, are widely available. Therefore, limited consumer access to 3DP hardware itself is becoming a concern of the past.

In the field of technology, once a piece of hardware reaches maturity, it often becomes evident that its software is the limiting factor of its potential. Additive manufacturing is no exception to this trend. There still remain a number of problems that 3DP software must solve. Additionally, when technology matures, it is also common for the software tools to become more important than the hardware itself. Most of the manufacturing process will usually happen in software and the user will spend most of the time using specialized design packages. This is exemplified by traditional paper printing: today, printing a poster is not typically as difficult as designing it.

Commercial and open source 3DP programs like Meshmixer, FreeCAD, OpenSCAD, Blender and ThinkerCAD are available. However, these were created with CAD in mind and then adapted to additive manufacturing. Specific problems related to additive manufacturing, like support generation, model orientation, and infill generation still demand the development of more specialized software.

1.1 Background

Traditionally, the digital manufacturing process has been subtractive. In this method, also known as *machining*, the material is removed in order to make the working piece. This process has the disadvantage of not being able to create all possible designs. For example, a design that contains inner structures enclosed by a hull cannot be created by machining. For this reason, in order to manufacture a complex working piece, the piece needs first to be segmented into smaller components that then may be machined individually. Moreover, this method introduces the disadvantage of needing to assemble the components.

By contrast, additive manufacturing has more expressive power, but has historically been more expensive. In additive manufacturing, the working piece is built layer upon layer, each being adhered to the previous one. Several 3DP technologies exist to create working pieces that span a wide range of materials, prices, qualities, and functionalities. We take the technological classifications from Horvath (2014) to divide the different technologies available today into three broad categories.

1.1.1 Selective deposition

In this set of technologies, the material is deposited only in the corresponding place. This is currently the most common technique available for 3DP and the *de facto* standard for hobbyists and home-owned printers. In this process, also known as Fusion Deposition Modeling (FDM), the thermoplastic material comes as a solid filament in a coil. First, the nozzle heats to the melting temperature of the plastic and starts extruding it. This creates a flexible plastic string that is deposited in rows to fill a layer. The plastics typically used in this process are polyactic acid (PLA) and acrylonitrile butadiene styrene (ABS). PLA has a lower melting temperature and can be extruded in thinner strings, which provides more detail. However, ABS offers more structural strength for the working

piece. It is also worth mentioning that the creation of filaments from other plastics is a subject of ongoing research. Prototypes developed to provide more flexibility, more strength, conductivity, and even florescence exist.

Since the term FDM is trademarked, the process is commonly referred to as Fused Filament Fabrication. An example of an FDM printer is depicted in Figure 1.1.



Figure 1.1. MakerBot Replicator 2TM printer by Makerbot industries. The image is by OhanaUnited - Own work, CC BY-SA 4.0, <http://commons.wikimedia.org/w/index.php?curid=54909778>.

1.1.2 Selective binding

In this technique, a powder material is scattered throughout the layer and is selectively bound on the part of the layer corresponding to the work piece by an agent. This has the advantage of not requiring support material, as the powder bed serves as support. Figure 1.2 shows a picture of the printing bed of one of these printers. Typical binding agents vary depending on the hardware, but they are usually activated by heat from a laser, or else are a liquid polymer. Typical powder materials include gypsum and metal. The powder is usually recycled. The quality of the piece is usually superior to that of FDM pieces. However, the process is more expensive. Also, because the powder creates a rough outer finish and because the bonds between layers are not especially strong, selective binding requires a post

process of curating, which involves some adhesive agent. Selective binding also allows dyes to be introduced to the process, which makes full color results possible. However, the resulting work pieces tend to be delicate, so the main application of this technique is for creating decorative pieces.



Figure 1.2. The ZPrinter 650TM by 3D Systems. The image is taken from www.3dnatives.com

1.1.3 Selective solidification

This family of printers uses a liquid material that is solidified by light. In this process, the building plate is covered by a thin layer of liquid. The light focuses on the membrane to solidify some of the liquid. Then, the platform is lowered, causing another layer of liquid to cover the first. The process repeats in this fashion until the piece is completed. The first 3D printers used this kind of technology. In these early printers, the material was a liquid photopolymer solidified by UV light in a process called stereolithography. The file format *.stl derives its name from this technology.

This approach is the most expensive type of additive manufacturing. However, it can give superior resolution and it is able to employ multiple types of printing materials. A printer of this type is shown in Figure 1.3.



Figure 1.3. The Form 1™ printer by Formlabs. The image is taken from www.3dhubs.com

1.2 Common Problems in additive manufacturing

In the field of digital manufacturing, multi-criteria optimizations are very common. Kolahan, Golmezerji, and Moghaddam (2012) describes problems in machining where several techniques are needed to optimize multiple criteria. Also, in Moshat, Datta, Bandyopadhyay, and Pal (2010) the authors describe a way to find the parameters to later use them in another optimization process.

Additive manufacturing remains an active field of research despite a myriad of advances in 3D printing. Gao et al. (2015) identifies a selection of problems that are currently the subject of substantial research in the field. First is the problem of mass production: even though additive manufacturing is an economical method for rapid prototyping, it typically proves expensive for mass production. Another issue of interest is the tradeoff between quality and speed when choosing an appropriate layer height. Finally, some research investigates the ways that variations in printing

materials can alter the physical properties of the piece: some materials are light but have low cohesion, while others are expensive and heavy but create a more durable piece. Gao et al. also highlight the difficulty of protecting intellectual property as 3DP becomes easier to use and more widely available.

Another survey paper, Oropallo and Piegl (2015), describes ten contemporary problems in 3DP. One of these is the issue of structural stability. If the inner structure of a layer changes, this affects the structural strength of the piece. Another problem is detecting when a model is impossible to manufacture (whether for lack of structural support or for another reason), which is known as the printability problem. Since models are commonly created in CAD programs that are not intended to produce designs for manufacture, the resulting pieces are not guaranteed to be printable. For example, if the pieces contain features too thin for the resolution, or if the piece has topological inconsistencies (if the model is not watertight), the printed piece may not be printable. Section 2.1 reviews work on this issue.

Oropallo and Piegl (2015) also identify determining the correct orientation of the piece as a key problem in 3DP. The orientation of the piece affects the quality of the surface, the piece's structural strength, and the speed of fabrication. However, Oropallo and Piegl also note that these goals can oppose each other.

According to Oropallo and Piegl, 3DP is divided into preprocessing, manufacturing, and postprocessing, and the speed of each subprocess is still an open problem. Preprocessing involves planning for manufacturing. The design of the piece and the printability assessment are two examples of preprocessing. Manufacturing can be optimized by better toolpath generation, better slicing, and optimizing the piece's orientation on the building plate. Finally, postprocessing involves problems like support removal, curation, and assembly. This work tries to improve the speed of 3DP during the manufacturing stage by modifying the orientation of the piece.

1.3 Statement of the problem

One of the disadvantages of rapid prototype design is the time of manufacturing. By changing the orientation of the piece with respect to the building plate, the manufacturing time can be modified. In previous research, several factors were taken into account when determining the most efficient orientation in terms of printing time.

In FDM, when each layer is fabricated, the printing head needs to travel between the different regions of the piece. When the head is outside of a printed region, the deposition of the material must stop. Then, it needs to restart when the head is inside of another printed region. Time is wasted not only when traveling in void regions, but also in the stop and restart of the plastic extraction. This factor has not been completely represented in previous attempts to optimize the orientation of a piece.

1.4 Significance

If a working piece needs to be produced en masse, even a few minutes eliminated from the manufacturing time can translate into considerable cost savings. The printing technologies currently available vary greatly and are still continuing to evolve. For example, changes might be due to a different printer, a different printing material, or a firmware update. Our algorithms have a way to reflect these changes and incorporate them into the optimization.

The algorithm is independent from the printing framework. It receives a mesh as an input and produces a possibly rotated mesh. This allows the separation of manufacturing and optimization. Once the optimization is found, printing can occur on a different computer. The optimal orientation remains valid even when the model is to be printed several times or in different scales.

1.5 Statement of purpose

The purpose of this project is to reduce the printing time of a piece by changing its orientation with respect to the building plate. The orientation of a model is a significant contributor to its printing time, but there are several factors that influence what the optimal orientation will be for a given model. Other research has correctly identified and optimized some of these factors, like the number of layers, the amount of support needed, etc. However, the time that the printing head wastes when traveling between connected components has been neglected. We propose the creation of an algorithm that incorporates this criterion in the optimization process.

We start by defining an optimization problem: finding the orientation of a piece that minimizes a function composed of four variables. The first three variables represent factors that have previously been proven to contribute to the printing time. We include one new variable to take into account the void path travel. Since the optimization problem is defined in a continuous space, we also describe several strategies to make the computation possible.

Since different printing technologies are affected differently by the aforementioned factors, we describe a method to calculate those factors for a giving printing framework. Therefore, is possible to calibrate our algorithm every time some piece of the framework changes. Finally, we implement the algorithm in software and make several experiments for a specific printer.

1.6 Research Question

Is it possible to increase the speed of 3DP by minimizing the void paths traveled by the nozzle?

1.7 Assumptions

This research uses the following assumptions:

- The models are accurately represented by watertight meshes.
- The input and output files are in Wavefront `obj` format, which contains geometrical information including the vertex position as well as the connectivity information to create faces.
- The meshes are triangular meshes.
- The software printer simulator produces results that are sufficiently accurate for the evaluation purposes of this study.
- The amount of time taken by the optimization is negligible compared to the total manufacturing time.

1.8 Limitations

This study is limited by the available technology, the time available to perform experiments, and the models available for testing. The following list enumerates these specific points.

- At the moment of performing the optimization, we need to make a discretization to analyze the slices and we need to impose a sampling delta close enough to the printer resolution.
- When we calculate the optimal weight, we use arbitrary models.
- The toolpaths for manufacturing the layers are not under our control.
- For our experiments, we use models free from copyright and available to the general public.
- We perform most parts of the experiments using a software simulator. For the actual manufacturing, we use only one FDM printer.

1.9 Delimitations

The delimitations for this study include:

- The number of models analyzed to determine the input parameters.
- The number of models used when comparing to the other method.
- The sampling $\Delta = 0.1$ used to determine the weights.
- The uniform scale for models used during manufacturing to reduce printing time.
- The quality of the surface, which is affected by printing a model in a different orientation. We decided to neglect this factor.

1.10 Definitions

In the broader context of thesis writing, we define the following terms:

3D printing The process of additive manufacturing.

Building direction The direction orthogonal to the plane where the layers are built in additive manufacturing. We use by convention, the vertical y direction.

Building plate A physical plate on which the working pieces are built. This plate is orthogonal to the building direction.

Layer The material that is deposited in a given plane on or above the building plane. Part of the slice that needs to be fabricated.

Model A geometric representation of the desired working piece. In concrete terms, this is a 3D mesh.

Orientation A rotation that is applied to the model, changing its position relative to the building plate before starting the manufacturing.

Slice The intersection of the interior of the mesh with the building plane.

Toolpath A directed curve that covers the area to be printed in each slice. A path to create a layer, given a slice.

Trapped Volume The empty volume that will be below a future layer of material.

Volume where a support structure may be necessary.

Watertight A non-self-intersecting polygonal surface that does not contain any discontinuities (holes). Formally, this is a piecewise continuous 2D manifold that fulfills the conditions of the Jordan–Brouwer separation theorem in \mathbb{R}^3

Working piece The product of the digital manufacturing.

1.11 Summary

This chapter gave a brief introduction to additive manufacturing technology as a whole. First, we reviewed the three main technologies for 3D printing. Then, we stated the problem of finding the best initial orientation for a model to manufacture. We explained the effects orientation can produce in the final working piece and why our research only addresses reduction of building time. We established the research framework for the rest of the document by posing the research question, assumptions, limitations, and delimitations. Finally, we defined concepts that will appear in the rest of the document.

CHAPTER 2. REVIEW OF RELEVANT LITERATURE

Additive manufacturing has been an industry tool for prototyping design since the 1980s. However, it was during the early 2010s, with the advent of low cost, FDM, plastic-based printers, that the field became important for the computer graphics (CG) community. This was the first time that 3D printing was available to the end consumer. The lack of software tools that could achieve high-quality results soon became obvious, however.

Circa 2010, the CG community started to innovate in this field, as most of the expertise required for additive manufacturing overlaps with that developed by CG researchers.

We reviewed the previous work, dividing the available research into three categories. The first category includes work focused on ensuring printability and on adding functionality to working pieces by altering the model. This also includes work that offers strategies for segmenting the model for manufacturing. The second category centers on support structures. This work describes how to generate supports, which strategies circumvent the need for them, and which novel applications support structures can be applied to. While some of the work applies to 3DP in general, we emphasized FMD printers in our research. The final group contains work that offers various strategies to optimize the building time of working pieces.

2.1 Structural properties

Volumetric representations of the input models have been used in Telea and Jalba (2011) to make voxel-based analyses and detect parts of the model that are not printable. FEM has been used in Stava, Vanek, Benes, Carr, and Měch (2012)

to detect areas of structural instability as well as to propose alterations to the model to correct the defects. Figure 2.1 depicts an example of the result of this process. Similarly, in Zhou, Panetta, and Zorin (2013), geometric analyses are able to create a map of weaknesses in a model quickly and accurately. Various solutions for these weaknesses have been developed. For instance, Lu et al. (2014) introduces honeycomb-like internal structures to a hollow model to increase the model's structural strength. Since this correction remains inside the surface of the model, the appearance of the model is not altered. One recent study, Zhang et al. (2015), describes a method for calculating internal support structures for the models by using the medial axis transform. Implementing the structural analysis in the design process instead of performing it separately after the piece is finished can result in greater structural integrity, as is described in the work of Xie, Xu, Yang, Guo, and Zhou (2015). A parametric representation of the surface that can also be used to design surfaces that are guaranteed to be self-supporting is discussed in Miki and Block (2015). Another recent study, Mirzendehdel and Suresh (2016), describes a topological optimization that is both light and also capable of supporting a predefined load. The study presents a method to fabricate such pieces, minimizing the support structures needed for manufacturing. Rigid body analysis of several plausible loads can be used to solve several FEM problems; the analysis is used to create a probability of failure map in Langlois, Shamir, Dror, Matusik, and Levin (2016).

Chen (2014) considers the deformation of the model after fabrication. The study describes the fabrication of a modified model that, once deformed by gravity, assumes the original desired shape. Another workaround involves fabricating a structure that approximates the exterior surface of the model in a similar fashion to produce a wireframe of the model. In Lorraine and Dick (2015), the authors find an exemplar pattern and reproduce it on the model's surface to ensure that the resulting surface is structurally sound. One example of their results appears in Figure 2.2.

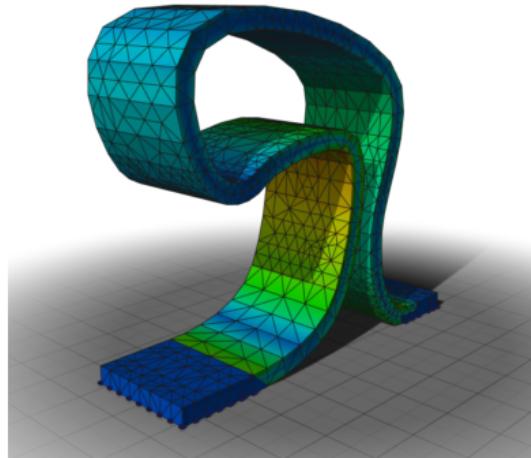


Figure 2.1. Analysis of stress using an FEM method to assess printability. Parts with high stress are more likely to collapse at manufacturing. The image is taken from Stava et al. (2012)



Figure 2.2. Approximating the model surface by exemplars, which are structurally sound. The image is taken from Lorraine and Dick (2015)

It is also possible to segment an object so that the separate pieces can be fabricated individually and assembled to form the original shape. This permits the printing of an object larger than the original printer chamber. Luo, Baran, Rusinkiewicz, and Matusik (2012) propose the first automatic method for segmenting large objects into pieces so that they can be assembled post-print. Vanek, Galicia, Benes, Měch, et al. (2014) builds on this methodology by demonstrating a way to pack the resulting pieces in order to minimize the number

of batches to print. Figure 2.3 shows this process. Chen, Lu, Hu, Cohen-or, and Chen (2015) contribute further by describing techniques for packing printed pieces more efficiently, thus optimizing the final packing result.

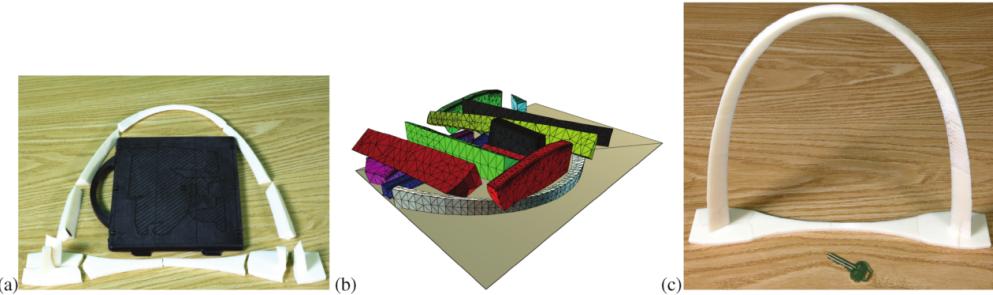


Figure 2.3. To print a model that is bigger than the printing plate (a) is possible to segment it in smaller pieces. Then, they can be tightly packed (b) to increase printing speed. Finally, assembling and gluing is needed to produce the final piece (c). This image is from the work of Vanek, Galicia, Benes, Měch, et al. (2014)

2.2 Supports generation

Structural support is identified as one of several criteria that can be altered by changing model orientation by Majhi, Janardan, Smid, and Schwerdt (1998). The same authors present a method to optimize support for convex models and a mathematical framework to minimize the support volume in 2D (or, to be more precise, the support area), as well as an optimization algorithm for minimizing the support volume for convex polyhedrons in Majhi, Janardan, Smid, and Gupta (1999).

Ilinkin et al. (2007) minimize the surface requiring support rather than the volume because the latter has greater impact in the final quality of the model. A decomposition of the model in pieces in order to avoid the need of support material is presented in R. Hu (2014). The study employs pyramidal shapes because they do

not need extra support to print and they are easy to assemble (as demonstrated in Figure 2.4).



Figure 2.4. A segmentation on a model in pyramids for eliminating the use of support.
This image is from the work of R. Hu (2014)

Support structure generation is typically addressed via one of two strategies. These are (1) based on the scaffolding used for creating buildings like the ones in Dumas, Lorraine, Hergel, and Lefebvre (2014) and (2) based on a tree-like structure that minimizes the material used in the support like Vanek, Galicia, and Benes (2014). Figure 2.5 shows these two approaches. We can also observe in this image that the support structures from both strategies are proportional to the trapped volume. Schmidt also proposes a tree-like structure generation technique Schmidt and Umetani (2014).

When using the techniques of tree-like support generation, In K. Hu, Zhang, and Wang (2016) the input model is altered to reduce the overhang area (i.e., the part of the model requiring support during manufacturing), thus reducing the need for structural support. Another part of the process that can be optimized is the placement of the supports in the overhang area. Yu-xin, Li-fang, Jian-kang, and Runyu (2015) propose a Poisson sampling to reduce the need of such points. Wang

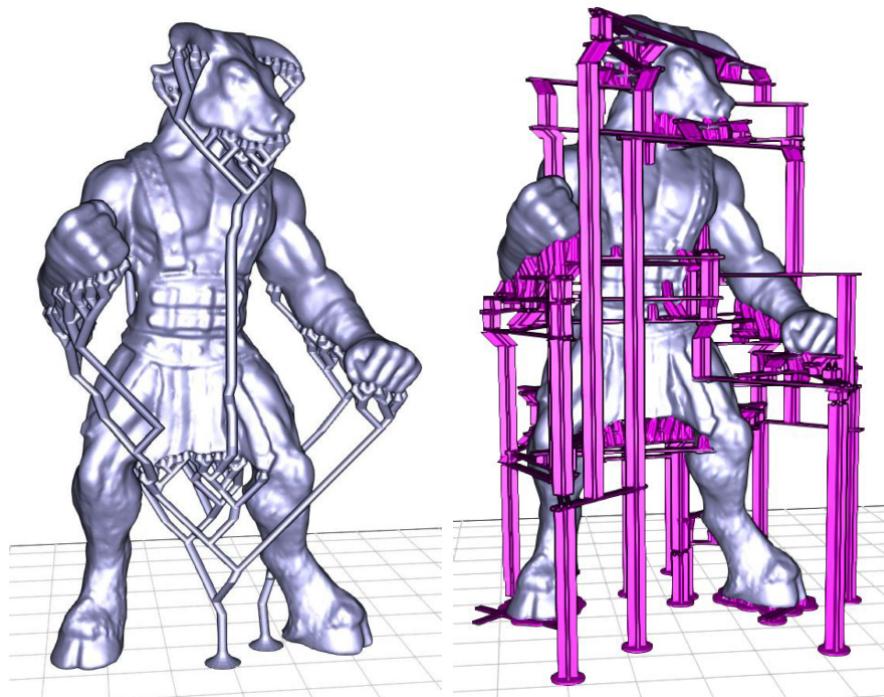


Figure 2.5. Two different support structures. The one on the left uses a tree-like structure that minimizes the wasted material. The one in the right uses a scaffolding structure which maximizes stability during manufacturing. The image is taken from Dumas et al. (2014)

et al. (2013) study the creation of an inner support structure not for the manufacturing process itself, but instead for adding structural soundness to a hollow model (as seen in Figure 2.6). Mirzendehdel and Suresh (2016) offer an optimization method that takes into account the need for supporting structures in the final model, rather than during the printing process alone.

2.3 Improving the building process

The early work of Schwerdt, Smid, Janardan, and Johnson (2003) recognizes that the orientation problem has many facets. In the study, the orientation is constrained to protect certain important parts of the model by disallowing them



Figure 2.6. This image shows inner supports for structural soundness of a hollow object. The image is taken from Wang et al. (2013)

from being overhangs in the printed piece. Another multi-criteria optimization that considers building time, part quality, and building material is that provided by Phatak and Pande (2012). The optimization method uses a genetic algorithm approach to perform the optimization. This is due to the high computational cost of computing these factors. The quality of the surface is addressed in Delfs, Tows, and Schmid (2015), which assigns a roughness factor to each facet of the object, then optimize the object's orientation with the help of an already-existing database of surface qualities. The research of Ahsan, Habib, and Khoda (2015) contributes greatly to this work. Ahsan et al. use a two-step optimization process to optimize first the building orientation, then the toolpath direction. Our work differs from that of Ahsan et al. (2015) insofar as the latter does not consider the distance between connected components. Ahsan et al. also assume that the toolpaths for fabrication lie always in parallel lines. The work of Wang, Zanni, and Kobbelt (2016) uses concepts from several of the aforementioned studies to optimize printing by segmenting the print model in several pieces. Because the outer surface of the piece does not require support and is almost aligned with the building direction, connectors can be added to the pieces and the assembly order can be computed. Alexa and Hildebrand (2017) assume that the machine is able to perform an

adaptive slicing. Thus, the thickness of each layer can vary. The orientation that can produce a slicing of minimum surface error and maximum speed is optimized. Their results are shown in Figure 2.7.

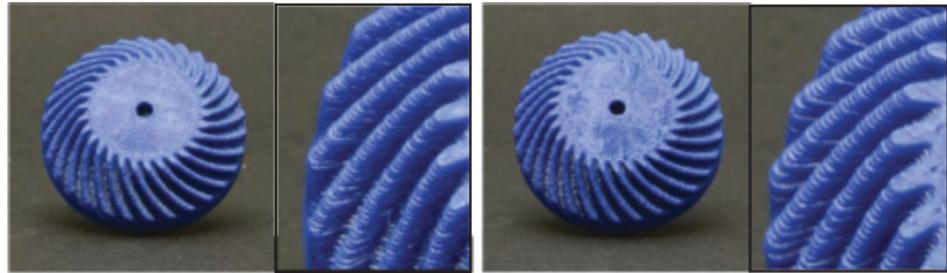


Figure 2.7. In the right, the piece is fabricated by an optimal adaptive slicing. The image in the left was fabricated by a traditional uniform slicing. The image is taken from Alexa and Hildebrand (2017)

Hon, Janardan, Schwerdt, and Smid (2003) provide one of the first studies that recognizes that toolpath generation should be optimized. The study assumes that such toolpaths are always in parallel lines. It attempts to optimize a major deposition direction by rotating the object around the building direction. A strategy of creating contour parallel toolpaths near the boundary of a layer and zig-zag patterns in the interior is discussed by G. Q. Jin, Li, and Gao (2013). Later work by Y. Jin, He, Fu, Gan, and Lin (2014) develops another strategy that calculates an optimal direction for creating parallel toolpath by using different weights for the speed and the quality of the layer. This study also takes into account the time spent traveling between connected components. Recent research by Zhao, Chen, Tu, and Chen (2016) favors a toolpath generation technique that uses Fermat spirals, which, due to their smooth curvature, have the advantage of preserving the object's surface quality as well as improving the speed of manufacturing. The method is powerful enough to fill a topologically-complicated region (for example, a region containing many random holes) with a single continuous curve as shown in Figure 2.8. An

analysis of how to plan the toolpath given a topologically-complicated surface appears in Lin et al. (2017), although the toolpath in the analysis is based on parallel infills. However, no studies have coupled the number of connected components with the distance between them in order to improve the printing time.

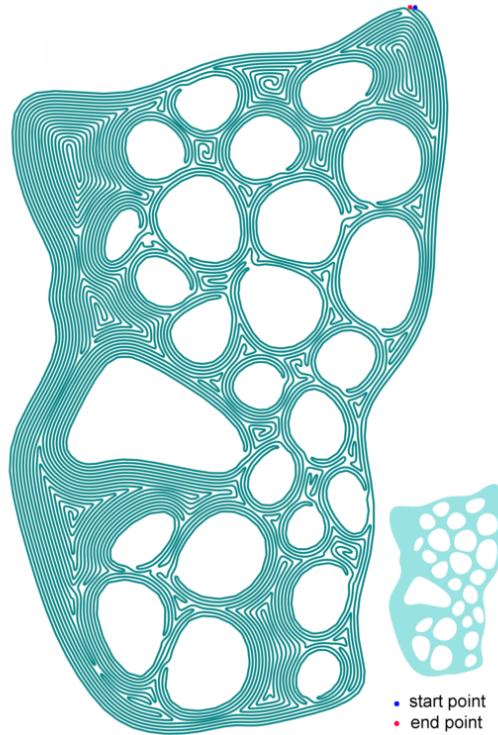


Figure 2.8. A slice is fabricated by a single path based on Fermat spirals. The image is taken from Zhao et al. (2016)

2.4 Summary

This chapter provided a review of relevant literature and divided the sources discussed into three groups. In Section 2.1 we examined first the issue of printability, then the ways to add functionality to a model, and finally the strategies to manufacture a piece bigger than the printing chamber.

Section 2.2 was concerned with support structures. We examined different approaches to generate structures, strategies to avoid the need for support, and several situations which employ supports not only for manufacturing.

Section 2.3 examined different methods that have been proposed to improve the manufacturing process. These works focused on the effects of changing the orientation and give strategies for toolpath generation.

CHAPTER 3. METHODOLOGY

The purpose of the algorithm in this study is to decrease the printing time for a given model. Because we assume that we do not have any control of the process once the manufacturing has started, we aim to change only the initial conditions of the manufacturing process. The manufacturing time is a function of both the general printing framework and the specific model of printer used. We change the initial orientation of the model with respect to the building plate to reduce cost. In other words, an optimal orientation is the one that, once applied to the model, produces the shortest manufacturing time for the piece.

Our algorithm operates by estimating the time of several orientations. In order to perform these estimations, it assumes that the time of manufacturing depends on four factors. The first three, total height, amount of support, and number of regions, have been already used by Ahsan et al. (2015) with the same objective. We find that with the inclusion of a new variable that takes into account the length of the void paths, the estimation is more accurate and the algorithm performs better.

Since the domain of the possible orientations is continuous, we perform a sampling to make the problem combinatorial. Section 3.2.1 explain this in greater detail. After sampling, we perform an exhaustive search on all the resulting orientations, keeping the ones with minimal cost (printing time). In this process, it is possible to have several optimal orientations.

Calculating the cost of each orientation requires the algorithm to slice the model and analyze each slice separately. This is analogous to actually simulating the printing and therefore computationally very expensive. Therefore, in Sections 3.2.2 and 3.3 we describe several strategies that the algorithm employs to estimate the cost to make the optimization more feasible.

As we mention in the beginning of this chapter, the time of printing is a function of the model and the printing framework. To account for the framework, the algorithm requires a set of parameters that weight each of the criteria according to the setup. In Section 3.4, we describe a methodology to obtain the parameters for a specific printing setup.

3.1 Overview

An overview of our system is shown in Figure 3.1. The input to our algorithm is a 3D model represented as a triangular watertight mesh such as an `obj` file. The output is its optimal orientation for printing represented as a pair of angles $(\alpha_x, \alpha_z)_{min}$ that correspond to the rotation that needs to be applied to the input model to position it in the optimal orientation.

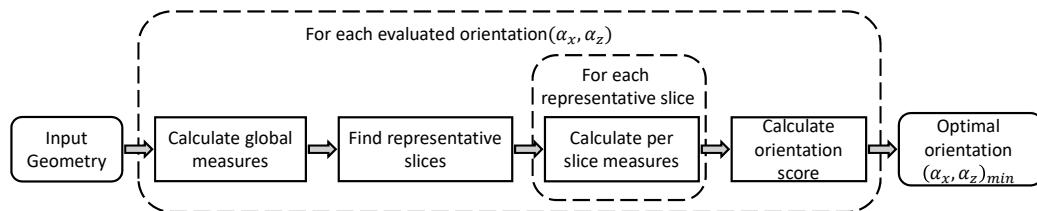


Figure 3.1. The optimization is in two main loops represented by the dotted lines: one that transverses all possible orientations and an inner loop that uses all the representative slices per orientation.

In the first step, we precompute a set of *evaluated orientations*. Because it would be unfeasible to evaluate the hypothetically infinite number of possible orientations, we assume that printing time does not change significantly for small changes in the object's rotation. Therefore, we discretize the continuous domain into a set of fixed orientations with a user-defined number of samples. The computation is then treated as a combinatorial problem, which allows us to perform an exhaustive search on all orientations (Section 3.2).

We consider four different criteria during the optimization. We use two global criteria, or criteria *per orientation* for each orientation of the object: the height of the object (denoted by H) and the trapped volume that requires support structures (denoted by V). In addition, we consider two local criteria or *per layer* criteria: the number of connected components C and their average distance D .

3.2 Optimization problem

The input object is located in the origin of the coordinate system and the printing axis is aligned with the y direction. The object can be rotated around two angles (α_x, α_z) to a new orientation. The angle $\alpha_x \in [0, 2\pi]$ represents the amount of rotation around the x axis and the angle $\alpha_z \in [-\frac{\pi}{2}, \frac{\pi}{2}]$ represents rotation around the z axis. By convention, these rotations are performed first around z and then around x .

Once the orientation is defined, a plane orthogonal to the building direction is swept by moving it by a small δ_y . At each stop, the intersection of model on the plane is calculated. This process is called slicing, and the intersection at each plane stop is a *slice*. Since the mesh is watertight, a slice will consist of a set of closed regions like the ones shown in Figure 3.2.

The *toolpath* dictates the movement of the nozzle in order to deposit material to fill the contours of the slice. Most printers work in a similar fashion to raster displays, in which the path consists of parallel lines. However, newer printers allow for greater control over the nozzle and are able to create paths concentric to the boundary of the regions. The *contour parallel* paths improve the quality of the final results. However, if the region is not convex, the path can be discontinuous.

All the toolpath generation techniques mention above have a common disadvantage. If there are more than one region in the slice, the nozzle needs to travel between regions, and it will need to stop and restart the material deposition several times whenever it travels between the regions. For this reason, it is always

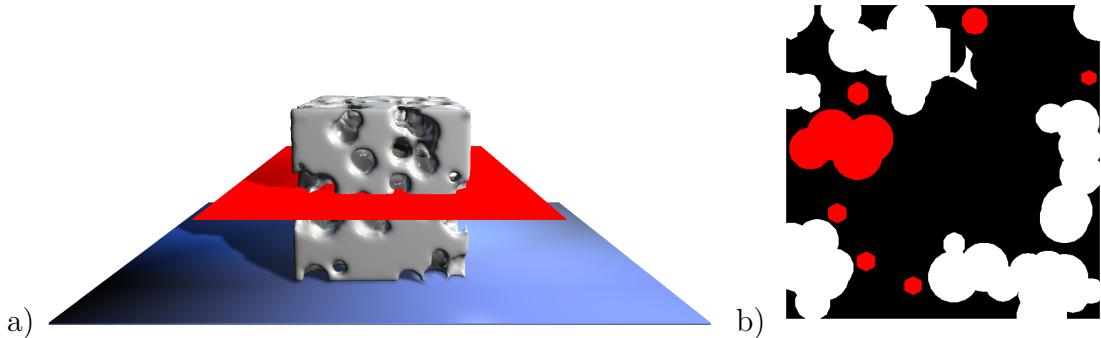


Figure 3.2. A slice is generated by the intersection of a plane orthogonal to the printing direction with the mesh. The slice (b) is composed of closed regions, some of which (shown in black) are filled by the material.

desirable to have fewer and smaller regions that are as close to each other as possible.

3.2.1 Sampling the solution space

In order to find a solution, we sample the spherical domain of pairs (α_x, α_z) that represent the points of a sphere. We start by approximating a sphere with an icosahedron. Then, we perform a recursive subdivision of triangles a given number of times (three in our experiments). Finally, we generate a sampling point at the projected centroid of each triangle on the sphere. This way, each point represents an equivalent area on the surface of the sphere. Since most of the models are stable at the orthogonal directions by design, we also add to our sampling the six orientations defined by the positive and negative directions of the three axis. Figure 3.3 shows the resulting regions for sampling.

Because of the optimization objective we use, a point in the north hemisphere of the sphere is *not* equivalent to the reflected point in the south hemisphere of the sphere.

The aim of optimization is to minimize a score function f defined as:

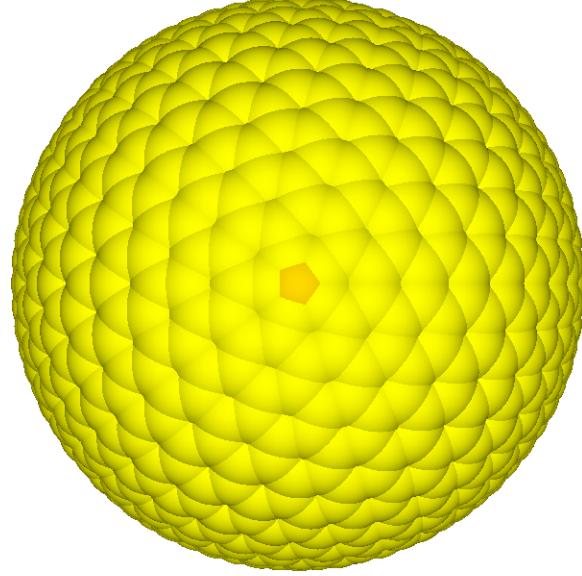


Figure 3.3. The spherical space is subdivided to ensure equal area on the bins (shown in yellow). The orange region represents one of the extra orientations added.

$$\min f(\alpha_x, \alpha_z) = w_h \cdot H + w_v \cdot V + \sum_{\forall i} r_i (w_d \cdot D + w_c \cdot C) \quad (3.1)$$

where H , V , D and C are our different criteria, all being functions of the orientation (α_x, α_z) . w represents the corresponding weight for the corresponding criteria. r_i represents the contribution of the i th representative slice.

Each of the components are explained in Section 3.3 in detail. It should suffice to say that H and V are calculated globally for each orientation. However, D and C are metrics per slide, so they are estimated for an orientation by measuring a set of representative slices that are in turn weighted for how much of the total model the given slice accounts for.

Since the criteria H , V , D and C are all calculated in different units and ranges, we perform a feature scaling

$$x' = \frac{x - \min(x)}{\max(x) - \min(x)} \quad (3.2)$$

before using them in Equation 3.1

3.2.2 Representative slices

Since the two per slice measures are functions of the regions, a slice is sampled every time this situation substantially changes (in other words, when a new region appears or an existing one disappears).

If an orientation is chosen and the plane is sweeping up from the tray in the building direction, a new region will appear when the plane reaches a valley in the surface defined by the mesh. Similarly, a region disappears when the plane reaches a peak of the surface. These are our critical points. An example appears in Figure 3.4 at left. Critical points are detected by comparing the angle of the surface normal to the printing direction. At the moment of comparison we give certain tolerance (5° in our implementation) to the normal, because in the mesh the normals are per vertex and are interpolated over the triangular faces.

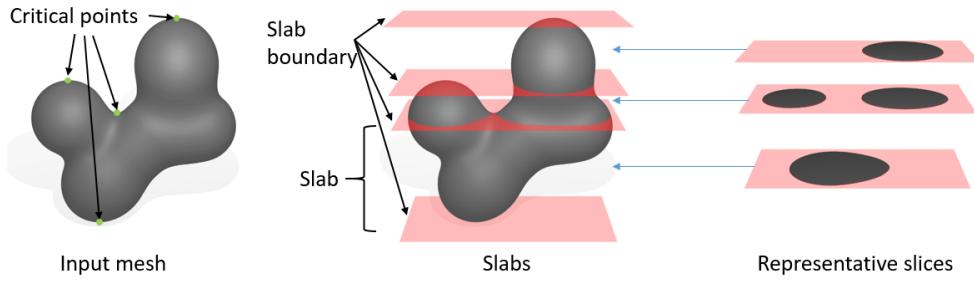


Figure 3.4. The critical points (right), define a set of boundaries that also define a slab (middle). We take just one representative slice per slab (left)

Once we have a list of the critical points p_i we order them according to their projection $h(p_i)$ in the building direction. It is usual for a mesh to have critical points to with the same $h(p_i)$. Again, in our implementation we discard the duplicates with 0.5mm tolerance. We create a plane orthogonal to the printing direction at each of the p_i . These define the slab's boundaries. This situation is

shown in Figure 3.4 at center, where the mesh is partitioned into slabs. The most important observation, however, is that on each slab, the numbers of regions is constant as shown in Figure 3.4 at right. Therefore, we take the plane at the middle of the slab to create a *representative slice* for each slab.

Since the slabs have different volumes, we calculate a weighting factor for each representative slice using the following formula.

$$r_i = \frac{h(p_{i+1}) - h(p_i)}{h(p_n) - h(p_0)}. \quad (3.3)$$

Where $h(p_i)$ and $h(p_{i+1})$ are the projections (or heights) of the building direction of the slice's lower and upper boundaries, and $h(p_0), h(p_n)$ are the projections of the first and last critical points.

3.3 Score Function

Typically, the height of the orientation H should be minimized, as printing in the vertical direction y is usually significantly slower than printing in the direction parallel to the printing plane. The trapped volume V should also be minimized because it requires additional support structures, which require time and material to build. Note that we do not examine the way these supports are constructed. Various methods exist for optimizing support structures, all of which can significantly affect the printing time. (See Section 2.2).

Having large numbers of disconnected components C is not beneficial because the printer needs to stop printing and move the printing head between them. If having disconnected components is inevitable, it is beneficial to minimize their distance D so that idle head movements are minimized.

3.3.1 Orientation score

Once the rotation defined by the orientation is applied to the mesh, we are able to calculate all the measures needed to get the score for the orientation. First we calculate an axis-oriented bounding box and use it to get the height H , or length of the box on the y axis. We want to minimize this height because the number of slices is considered the most influential factor that contributes to the printing time, and this is proportional to H .

In plastic FDM plastic based printers, another important factor is the number and size of support structures required to print. The presence of supports can affect the result in several ways: the time of manufacturing, since they need to be removed after the printing; the quality of the surface, because the surface is damaged at the time of removal; and the amount of wasted material, since the supports are discarded at the end.

Since the shape of the supports, and hence the exact amount of the material, depends on the software used to operate the printer, we use the strategy examined in Majhi, Janardan, Schwerdt, Smid, and Gupta (1999) to minimize the trapped volume V . This should be always proportional to the wasted material and the time used for printing the supports. See Figure 3.5

3.3.2 Slice score

A slice is composed of a series of line segments. Since the mesh is triangular, we can transverse all the faces. If in any triangle, one point is on a different side of the plane $h(p_i)$ with respect of the other two, we calculate the intersection of the triangle with the plane, which is the segment. Before we can obtain the components of the slice score, we need to partition the segments in contours. A contour is an ordered set of segments in which one segment starts where the previous one ends. A closed contour is a contour in which the last segment ends at the same point where the first started. Since we use a watertight mesh and the segments are results of the

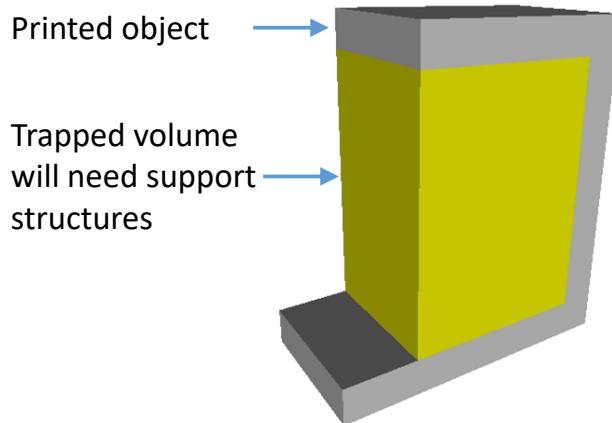


Figure 3.5. The material used for creating the supports for printing the gray object in this orientation is proportional to the trapped volume shown in yellow.

intersection of a plane with the mesh, a partition of the segments into closed contours always exists. See Figure 3.2 (b). In order to make the partition, we grow the contours. We start with one segment and we look for segment that have one common end point, when we found it we add it to the contour. We continue adding segments until, the contour closes. We use the term *region* for each of these closed contours.

Depending on the topology of the original mesh, a region on a slice may be contained inside of another one. For example, if a region contains a hole, the hole itself is also a region. If a region needs to be filled during the printing, we say that has *positive area*. In contrast, for regions where we don't need to fill the slice, we say that have *negative area*. Figure 3.6 shows an example.

In the our implementation, we compare each pair of contours, looking for containment. On the given conditions, one region A contains another one B , If all the points of B are inside of A . Once we made all comparisons, we look for the regions that are not contained by any other, we mark them as positive. Then, we look for regions contained only by a positive region, we mark them as negative.

Then, we look for regions contained only by a negative region, we mark them as positive. We continue in similar fashion until all regions are marked.

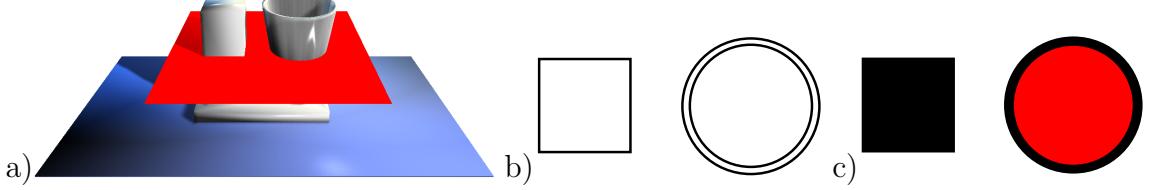


Figure 3.6. A model is sliced (a) and a set of contours is generated (b). The contours define positive regions that require filling (black) and negative regions (red) that are to be left empty (c).

We are interested in the number of positive area regions C in the representative slices. This quantity is called contour plurality in Ahsan et al. (2015).

Given two regions, we define the distance between them as the minimum distance between any point on one region to any point on the other region. Our last component D is the average distance between all positive regions of the slice.

3.4 Determining the corresponding weights

The objective function (3.1) has a set of user weights $w = (w_h, w_v, w_d, w_c)$ that must be specified before optimization. These parameters depend on the printer and other computational factors including the slicer and the path planning for the printing. The interaction of these parameters is not yet fully understood; they have been set as fixed values in Ahsan et al. (2015). We discuss a method to determine these values experimentally for any printer setting. Then, we report the values for the MakerBot Replicator+.

We evaluated $k = 20$ objects with widely-varying shapes and topologies (Figure 3.7). To do this, we sampled each parameter's space with sampling distance $\Delta = 0.1$. Because we had four parameters to determine, we created $n = 1/\Delta + 1$

samples and generated all possible n^4 tuples of (w_h, w_v, w_v, w_c) ; these led to $\approx 14k$ ordinations to be evaluated.

For each model, we found the optimal orientation for each set of weights. However, there were some duplicates. For example, all tuples (w_h, w_v, w_v, w_c) and $(\lambda w_h, \lambda w_v, \lambda w_v, \lambda w_c)$ for some λ resulted in the same orientation.

It would not have been feasible to print several thousands of objects to measure their printing times. Thus, after we found the weights, we used the printer simulator Printrun¹. The program allowed us emulate the settings of a particular printer to estimate the time that model would take to print for each orientation. We then recorded the orientation with the minimal time reported by the software. These were the optimal parameters for that particular model.

We repeated the process for each model, taking the arithmetic median of all the optimal tuples to obtain the printer parameters.

The resulting values for the normalized weights in our settings are:

$w_h = 0.34$, $w_v = 0.24$, $w_d = 0.11$, and $w_c = 0.31$.

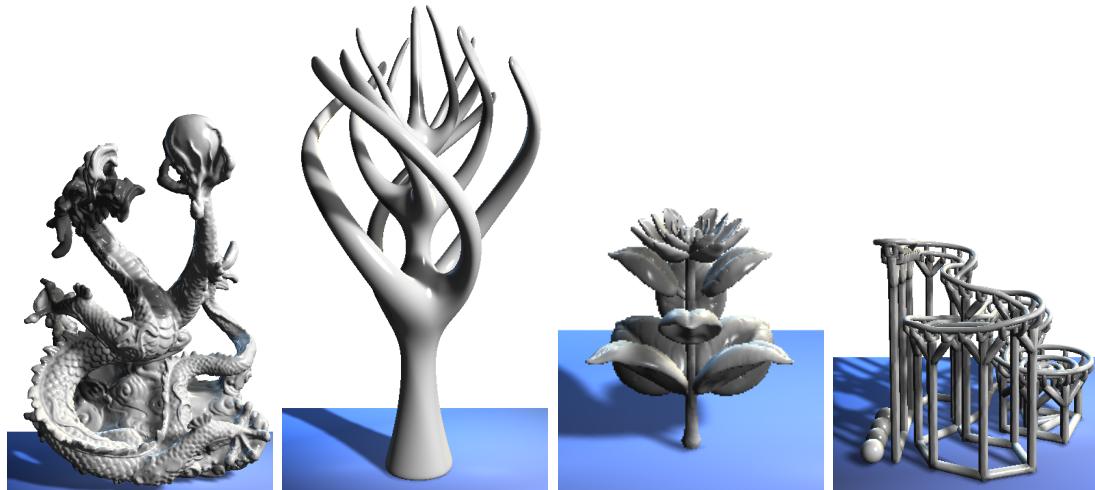


Figure 3.7. Some of the objects used to estimate the weights.

¹www.pronterface.com

3.5 Summary

This chapter provided a top down explanation of the proposed software pipeline. We started with a very general explanation of the optimization in Section 3.2, eventually describing the optimization problem. Then, we explained how we obtain the score for each orientation, which is divided into two general terms: per orientation and per slide. Next, we examined the individual terms of the objective function to explain how they are calculated. Finally, we also explained how to calculate w 's parameters for the equation 3.1 given a particular manufacturing setup.

CHAPTER 4. EVALUATION

In order to evaluate our solution, we compared it to the method of Ahsan et al. (2015). Since the authors describe a two step optimization and the second step involves how to orientate the object to produce better toolpaths, we only compared our method to the first step, which calculates the same orientation as ours.

One central finding of our work is the collection of weights for optimal orientation on the MakerBot Replicator+ 3D printer. In Ahsan et al. (2015), however, the authors do not report optimal weights for their algorithm. Thus, to make the comparison fair, we repeated the optimal weight-finding procedure from Section 3.4 using their criteria. This led to the set of weights of $w_h = 0.39$, $w_v = 0.26$, and $w_c = 0.35$. We then compared our algorithm with the Ahsan et al. algorithm by using optimal weights for both.

We simulated printing 953 models from the SHREC 2015 dataset¹. The repository contains 1,200 models, but we omitted those that were not printable.

We simulated the printing in the orientations obtained by the two weight sets: once for our algorithm and once for the Ahsan et al. (2015) algorithm (we report the printing time ratio of both methods in Figure 4.1). If the ratio was equal to one, both algorithms converged to the same orientation or an orientation that required the same printing time. Ratios greater than one suggested that the distance of the connected components in the slice should be considered because this converges to a faster solution. In other words, ratios greater than one suggested our optimization method improved printing speed. For some objects, the speed increase was as high as 45%. Ratios less than one suggested the opposite: that the extra criteria in our optimization decreased performance. The worst case for our algorithm was a 50% performance hit.

¹<http://www.itl.nist.gov/iad/vug/sharp/contest/2015/Range/data.html>

Overall, 62% of the cases of both algorithms arrived at the same orientation. The results also showed that considering the traveling distance can lead to worse performance in 23% of the cases. However, for the remaining 15%, considering the distance of the connected components led to a faster solution.

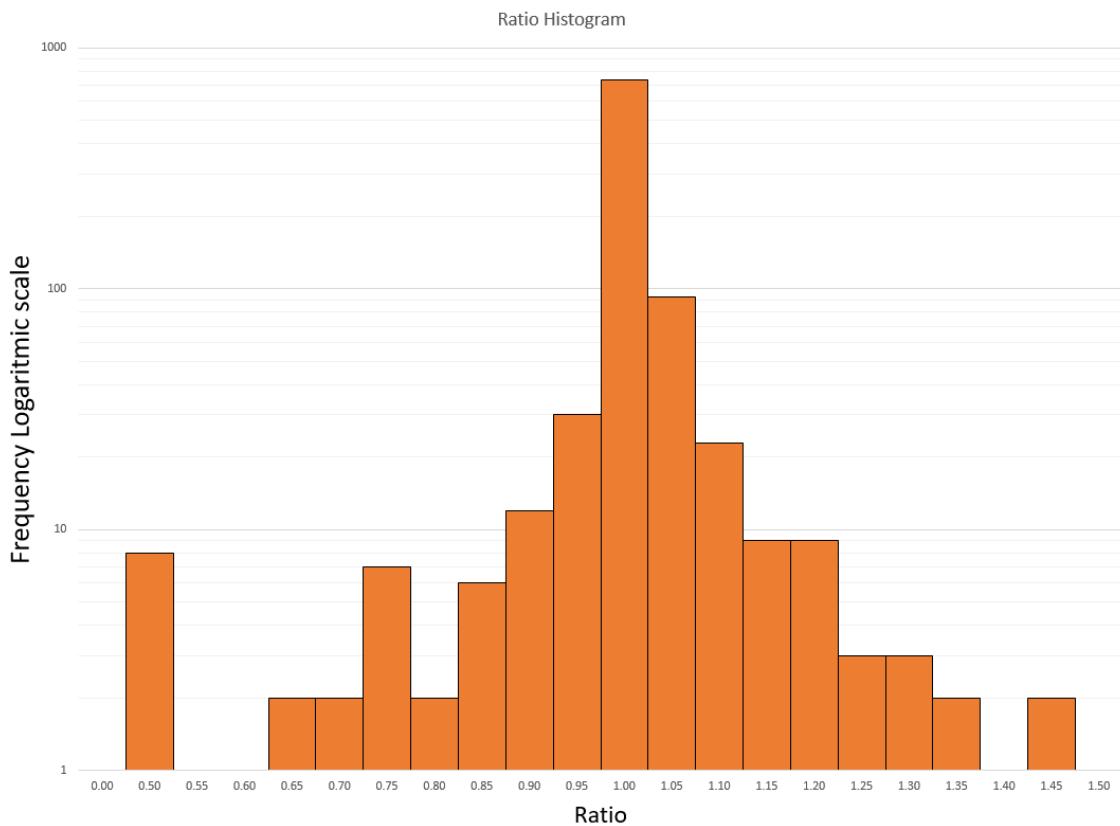


Figure 4.1. Histogram of the ratios of printing times for 954 objects. A ratio greater than one suggests that using the four criteria is better; smaller than one suggests that using only three criteria (the method suggested by Ahsan et al. (2015)) is better. Note that the y axis uses a logarithmic scale.

4.1 Outliers

After all tests were performed, we (physically) printed the three best and three worst objects from both sets and measured the printing time for both cases.

Table 4.1 shows the actual printing time and the printed objects in their orientations. These printings confirm our findings.

Figure 4.2 shows one of the outliers in a default position (as specified by its creator) as well as in the optimal orientation detected via current criteria. The orientation in Figure 4.2 a) is less suitable because it gives the object a significantly greater height than the object has in Figure 4.2 b). However, the reduction in wasted printing head movement makes the first orientation take only 69% of the time of the second.

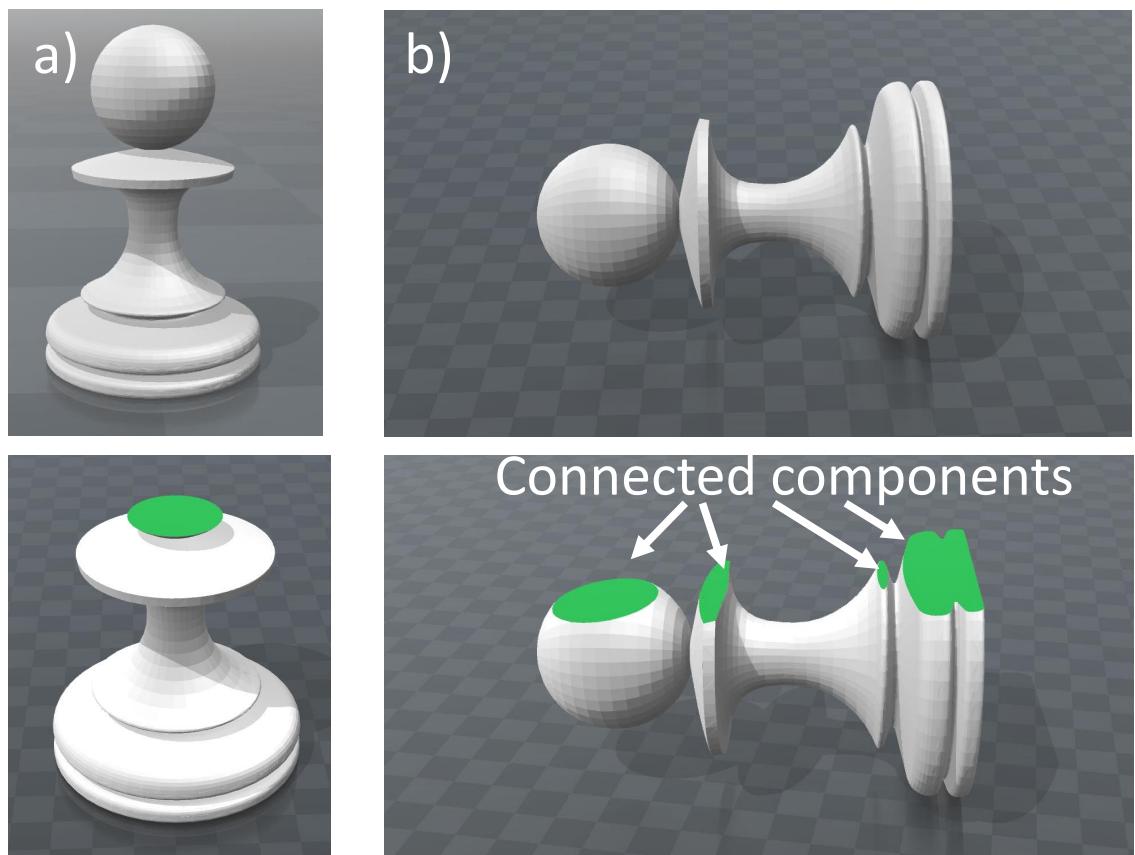


Figure 4.2. An object in an orientation that minimizes the number of connected components (a) and in an orientation that considers the height of the object (b). The first orientation takes 69% of the second's printing time.

The cases that do not show speed increases for our method occurred most likely from underestimating V in our experiments. This could suggest that thin, elongated objects that need a support and do not have a clear difference in C were not accurately represented in the determination of the optimal weights.

Although the actual printing and simulated time varied significantly (probably because of some systematic error in the simulation software), the ratios of the measured times agreed for the simulated and the measured printing times. Thus, we believe the simulation can be used to approximate the actual printing processes with accuracy.

4.2 Summary

In this chapter, we explained that the best evaluation strategy is to compare our system to the method offered by Ahsan et al. (2015). We then discussed how can we equalize the experiment to make this comparison fair and we presented the experiment's results. At the end of the chapter, we also analyzed the outliers of our experiment in detail in an attempt to predict the most- and least-suitable situations for using our method.

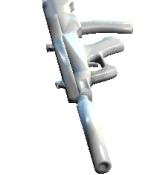
Our method			Method Ahsan et al. (2015)			Comparison	
	Time in seconds		Time in seconds			Ratio	
Orientation	Actual	Simulated	Orientation	Actual	Simulated	Measured	Simulated
Prefers distance of connected components							
	131	809		288	1434	2.19	1.77
	953	5847		1391	8418	1.45	1.44
	2609	10116		3804	16492	1.46	1.63
Does not prefer distances of connected components							
	671	1824		321	423	0.48	0.23
	1180	1327		737	224	0.62	0.17
	904	2263		605	782	0.67	0.35

Table 4.1.

Printing times for the best and worst cases that considered or did not consider the distance of connected components.

CHAPTER 5. CONCLUSIONS

We have developed a novel algorithm for finding a printing orientation for a 3D object that minimizes its printing time. Our key observation is that the time spent traveling over empty areas of the printed slice can be further minimized. We have incorporated this criterion into an existing optimization that also considers the object height, the trapped volume, and the number of connected components. Furthermore, we have provided a systematic method to adapt the optimization so it can represent any 3DP setup. This is archived by finding optimal w in equation 3.1 for a given printer. In this comparison, our method demonstrates printing time decreases as high as 45% .

The main limitation of our work derives from the way the weights are found. First, the representative model cannot contain enough variability. Second, the calibrations depend on two samplings: k and n . Obviously, if we increase either of these, our results should improve. However, the most important limitation comes from the fact that we used simulation software that needs to be carefully calibrated for an existing printer. The calibration, as well as the measured values, can vary depending on many unknown variables. It would be preferable to actually print all the objects and measure their printing time, but this is time- and cost-prohibitive. Our algorithm also uses several space discretizations (the orientations and the slicing process) that may have an effect on the final results. Moreover, the choice of the representative slice from the slab could also affect the results.

One avenue for future work would be to provide finer levels of granulation for the trapped volume. We consider its minimization to be an important factor. However, various approaches for building support structures already exist (Dumas et al. (2014); Huang, Fu, Wei, and Hu (2015); Majhi, Janardan, Schwerdt, et al. (1999); Mirzendehdel and Suresh (2016); Vanek, Galicia, and Benes (2014)). The

time needed to build support structures can significantly vary, affecting the overall printing time. Nonetheless, including them in the optimization could produce richer data.

5.1 Future work

One of the principal limitations of this study lies in our lack of control over toolpath generation. Very little research examines this; Zhao et al. (2016) is one of few examples. However, the authors focus only on one slice.

A future goal should be to incorporate a toolpath generation approach to our optimization. This will also mean that the quantity D should be replaced with a more effective measure of the unnecessary traveling of the printing head.

Another avenue should be to analyze slice coherence. This will also depend on the resolution of the printer and, again, our ability to control toolpath generation.

Finally, our experiment shows that the optimal orientation is very sensitive to the input model. Thus, it could be interesting to research how a pure learning-based approach compares to methods like ours, which try to optimize based on geometry and the physical conditions of the manufacturing process.

LIST OF REFERENCES

LIST OF REFERENCES

- Ahsan, A. N., Habib, A., & Khoda, B. (2015). Resource based process planning for additive manufacturing. *Computer-Aided Design*. Retrieved from <http://dx.doi.org/10.1016/j.cad.2015.03.006> doi: 10.1016/j.cad.2015.03.006
- Alexa, M., & Hildebrand, K. (2017). Optimal Discrete Slicing. *ACM Trans. Graph.*, 36(1).
- Chen, X. (2014). An Asymptotic Numerical Method for Inverse Elastic Shape Design. *ACM Transactions on Graphics*, 33(4). doi: 10.1145/2601097.2601189
- Chen, X., Lu, L., Hu, R., Cohen-or, D., & Chen, B. (2015). Dapper : Decompose-and-Pack for 3D Printing. *ACM Transactions on Graphics (TOG)*, 34(6), 213. doi: 10.1111/cgf.12811
- Delfs, P., Tows, M., & Schmid, H. J. (2015). Optimized build orientation of additive manufactured parts for improved surface quality and build time. *Additive Manufacturing*, 12, 314–320. doi: 10.1016/j.addma.2016.06.003
- Dumas, J., Lorraine, U. D., Hergel, J., & Lefebvre, S. (2014). Bridging the Gap : Automated Steady Scaffoldings for 3D Printing. *Siggraph 2014*, 33(4), 98–108. doi: 10.1145/2601097.2601153
- Gao, W., Zhang, Y., Ramanujan, D., Ramani, K., Chen, Y., Williams, C. B., ... Zavattieri, P. D. (2015). The status, challenges, and future of additive manufacturing in engineering. *Computer-Aided Design*. Retrieved from <http://www.sciencedirect.com/science/article/pii/S0010448515000469> doi: 10.1016/j.cad.2015.04.001
- Hon, M. C., Janardan, R., Schwerdt, J., & Smid, M. (2003). Minimizing the total projection of a set of vectors, with applications to layered manufacturing. *CAD Computer Aided Design*, 35(1), 57–68. doi: 10.1016/S0010-4485(01)00175-0
- Horvath, J. (2014). *Mastering 3d printing*. Berkeley, CA, USA A Press.
- Hu, K., Zhang, X., & Wang, C. C. L. (2016, Aug. 2015). *Direct Computation of Minimal Rotation for Support Slimming*. IEEE.
- Hu, R. (2014). Approximate Pyramidal Shape Decomposition. *ACM Transactions on Graphics*, 33(6), 213–225.

- Huang, S.-s., Fu, H., Wei, L., & Hu, S.-M. (2015). Support Substructures: Support-Induced Part-Level Structural Representation. *IEEE Transactions on Visualization and Computer Graphics*, 26(26), 1–1. Retrieved from <http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=7226860> doi: 10.1109/TVCG.2015.2473845
- Ilinkin, I., Janardan, R., Smid, M., Johnson, E., Castillo, P., & Schwerdt, J. (2007). Heuristics for estimating contact area of supports in layered manufacturing. *Journal of Experimental Algorithms*, 11(1), 1.6. doi: 10.1145/1187436.1210589
- Jin, G. Q., Li, W. D., & Gao, L. (2013). An adaptive process planning approach of rapid prototyping and manufacturing. *Robotics and Computer-Integrated Manufacturing*, 29(1), 23–38. doi: 10.1016/j.rcim.2012.07.001
- Jin, Y., He, Y., Fu, J., Gan, W., & Lin, Z. (2014). Optimization of tool-path generation for material extrusion-based additive manufacturing technology. *Additive Manufacturing*, 14, 32 - 47. Retrieved from <http://www.sciencedirect.com/science/article/pii/S2214860414000098> (Inaugural Issue) doi: <http://doi.org/10.1016/j.addma.2014.08.004>
- Kolahan, F., Golmezerji, R., & Moghaddam, M. A. (2012, 1). Multi objective optimization of turning process using grey relational analysis and simulated annealing algorithm. In *Mechanical and aerospace engineering, icmae2011* (Vol. 110, pp. 2926–2932). Trans Tech Publications. doi: 10.4028/www.scientific.net/AMM.110-116.2926
- Langlois, T., Shamir, A., Dror, D., Matusik, W., & Levin, D. I. W. (2016, November). Stochastic structural analysis for context-aware design and fabrication. *ACM Trans. Graph.*, 35(6), 226:1–226:13. Retrieved from <http://doi.acm.org/10.1145/2980179.2982436> doi: 10.1145/2980179.2982436
- Lin, Z., Fu, J., Sun, Y., Gao, Q., Xu, G., & Wang, Z. (2017). Non-retraction toolpath generation for irregular compound freeform surfaces with the lkhs tsp solver. *The International Journal of Advanced Manufacturing Technology*, 1–15. Retrieved from <http://dx.doi.org/10.1007/s00170-017-0247-8> doi: 10.1007/s00170-017-0247-8
- Lorraine, D., & Dick, C. (2015). By-Example Synthesis of Structurally Sound Patterns. *ACM Transactions on Graphics*, 34(4), 1–12.
- Lu, L., Sharf, A., Zhao, H., Wei, Y., Fan, Q., Chen, X., ... Baoquan, D. C.-o. (2014). Build-to-Last : Strength to Weight 3D Printed Objects. *ACM Transactions on Graphics*, 33(4), 1–10. doi: 10.1145/2601097.2601168
- Luo, L., Baran, I., Rusinkiewicz, S., & Matusik, W. (2012). Chopper. *ACM Transactions on Graphics*, 31(6), 1. Retrieved from <http://dl.acm.org/citation.cfm?doid=2366145.2366148> doi: 10.1145/2366145.2366148
- Majhi, J., Janardan, R., Schwerdt, J., Smid, M., & Gupta, P. (1999). Minimizing support structures and trapped area in two-dimensional layered manufacturing. *Computational Geometry*, 12(3-4), 241–267. doi: 10.1016/S0925-7721(99)00003-6

- Majhi, J., Janardan, R., Smid, M., & Gupta, P. (1999). On some geometric optimization problems in layered manufacturing. *Computational Geometry*, 12(3-4), 219–239. doi: 10.1016/S0925-7721(99)00002-4
- Majhi, J., Janardan, R., Smid, M., & Schwerdt, J. (1998). Multi-criteria geometric optimization problems in layered manufacturing. *Proceedings of the fourteenth annual symposium on Computational geometry - SCG '98*, 19–28. Retrieved from [http://dl.acm.org/citation.cfm?id=276887\\$&delimiter=026E30F&nhttp://portal.acm.org/citation.cfm?doid=276884.276887](http://dl.acm.org/citation.cfm?id=276887$&delimiter=026E30F&nhttp://portal.acm.org/citation.cfm?doid=276884.276887) doi: 10.1145/276884.276887
- Miki, M., & Block, P. (2015). Parametric Self-supporting Surfaces via Direct Computation of Airy Stress Functions. *ACM Transactions on Graphics (TOG)*, 34(4), 89:1—89:12. Retrieved from <http://dl.acm.org/citation.cfm?id=2766888> doi: 10.1145/2766888
- Mirzendehdel, A. M., & Suresh, K. (2016). Support structure constrained topology optimization for additive manufacturing. *Computer-Aided Design*. Retrieved from <http://linkinghub.elsevier.com/retrieve/pii/S0010448516300951> doi: 10.1016/j.cad.2016.08.006
- Moshat, S., Datta, S., Bandyopadhyay, A., & Pal, P. (2010). Optimization of cnc end milling process parameters using pca-based taguchi method. *International Journal of Engineering, Science and Technology*, 2(1), 95–102.
- Oropallo, W., & Piegl, L. a. (2015). Ten challenges in 3D printing. *Engineering with Computers*. Retrieved from <http://link.springer.com/10.1007/s00366-015-0407-0> doi: 10.1007/s00366-015-0407-0
- Phatak, A. M., & Pande, S. S. (2012). Optimum part orientation in Rapid Prototyping using genetic algorithm. *Journal of Manufacturing Systems*, 31(4), 395–402. Retrieved from <http://dx.doi.org/10.1016/j.jmsy.2012.07.001> doi: 10.1016/j.jmsy.2012.07.001
- Schmidt, R., & Umetani, N. (2014). Branching Support Structures for 3D Printing. *ACM SIGGRAPH 2014 Conference Proceedings, 10-14th August*, 9, 1. doi: 10.1145/2619195.2656293
- Schwerdt, J., Smid, M., Janardan, R., & Johnson, E. (2003). Protecting critical facets in layered manufacturing: Implementation and experimental results. *CAD Computer Aided Design*, 35(7), 647–657. doi: 10.1016/S0010-4485(02)00090-8
- Stava, O., Vanek, J., Benes, B., Carr, N., & Měch, R. (2012). Stress relief. *ACM Transactions on Graphics*, 31(4), 1–11. doi: 10.1145/2185520.2335399
- Telea, A., & Jalba, A. (2011). Voxel-Based Assessment of Printability of 3D. In *Voxel-based assessment of printability of 3d* (pp. 393–404). Springer Berlin Heidelberg.

- Vanek, J., Galicia, J. a. G., & Benes, B. (2014). Clever Support : Efficient Support Structure Generation for Digital Fabrication. *Symposium on Geometry Processing*, 33(5). doi: 10.1111/cgf.12437
- Vanek, J., Galicia, J. a. G., Benes, B., Měch, R., Carr, N., Stava, O., & Miller, G. S. (2014). PackMerger: A 3D Print Volume Optimizer. *Computer Graphics Forum*, 00(00), n/a–n/a. Retrieved from <http://doi.wiley.com/10.1111/cgf.12353> doi: 10.1111/cgf.12353
- Wang, W., Wang, T. Y., Yang, Z., Liu, L., Tong, X., Tong, W., ... Liu, X. (2013). Cost-effective printing of 3D objects with skin-frame structures. *ACM Transactions on Graphics*, 32(6), 1–10. Retrieved from <http://dl.acm.org/citation.cfm?doid=2508363.2508382> doi: 10.1145/2508363.2508382
- Wang, W., Zanni, C., & Kobbelt, L. (2016). Improved Surface Quality in 3D Printing by Optimizing the Printing Direction. *Computer Graphics Forum*, 35(2).
- Xie, Y., Xu, W., Yang, Y., Guo, X., & Zhou, K. (2015). Agile structural analysis for fabrication-aware shape editing. *Computer Aided Geometric Design*, 35–36, 163–179. Retrieved from <http://linkinghub.elsevier.com/retrieve/pii/S0167839615000424> doi: 10.1016/j.cagd.2015.03.019
- Yu-xin, M., Li-fang, W., Jian-kang, Q., & Runyu, W. (2015). An Optimized Scheme to Generating Support Structure for 3D printing. In *Image and graphics* (Vol. 1, pp. 571–578). Springer International Publishing. Retrieved from http://link.springer.com/chapter/10.1007/978-3-319-21978-3_50 doi: 10.1007/978-3-319-21969-1
- Zhang, X., Xia, Y., Wang, J., Yang, Z., Tu, C., & Wang, W. (2015). Medial axis tree: an internal supporting structure for 3D printing. *Computer Aided Geometric Design*, 35–36, 149–162. Retrieved from <http://linkinghub.elsevier.com/retrieve/pii/S0167839615000357> doi: 10.1016/j.cagd.2015.03.012
- Zhao, H., Chen, Y., Tu, C., & Chen, B. (2016). Connected Fermat Spirals for Layered Fabrication. *ACM Transactions on Graphics (proceedings of ACM SIGGRAPH)*, 35(4), 100:1—100:10. Retrieved from <http://doi.acm.org/10.1145/2897824.2925958> doi: 10.1145/2897824.2925958
- Zhou, Q., Panetta, J., & Zorin, D. (2013). Worst-case Structural Analysis. *ACM Trans. Graph.*, 32(4), 137:1–137:12. Retrieved from [http://doi.acm.org/10.1145/2461912.2461967\\$>\deliminter"026E30F\\$>nhttp://dl.acm.org/ft{ }gateway.cfm?id=2461967{ }type=pdf](http://doi.acm.org/10.1145/2461912.2461967$>\deliminter) doi: 10.1145/2461912.2461967

VITA

VITA

Jorge Antonio García Galicia

Education

PURDUE UNIVERSITY*West Lafayette, IN***PhD** in Technology*2012 - 2017*

Purdue Polytechnic Institute, Computer Graphics Department

Thesis: Optimizing part orientation for additive manufacturing using connected components

NATIONAL AUTONOMOUS UNIVERSITY OF MEXICO*México City***Master** of Science in Computer Science*2008 - 2011*

Institute of Applied Mathematics and Systems

Thesis: Smooth visualization of surfaces obtained by boundary tracking applied to discrete volumes

NATIONAL AUTONOMOUS UNIVERSITY OF MEXICO*México City***Bachelor** of Science in Applied Mathematics and Computer Science*2000 - 2005*

School of Higher Studies Acatlán

Thesis: Graphic modeling of a pneumatic body with differential equations using OpenGL

Professional and Academic Experience

Adobe Systems Incorporated*San Francisco, CA*

Research Intern, Procedural Image Group

2016

Nvidia Corporation	<i>Santa Clara, CA</i>
Software Developer Intern, OpenGL Driver Team	<i>2015</i>
Nvidia Corporation	<i>Santa Clara, CA</i>
Software Developer Intern, DirectX Driver Team	<i>2014</i>
Purdue University	<i>West Lafayette, IN</i>
Teaching and Research Assistant	<i>2012 - 2017</i>
National Autonomous University of Mexico	<i>México City</i>
Lecturer and Research Assistant	<i>2008 - 2012</i>

Awards and other nominations

Polytechnic Institute Summer Research Grant	<i>2017</i>
Scholarship for intentional graduate studies, CoNaCyT	<i>2012 - 2017</i>
Senator, The Purdue Graduate Student Senate	<i>2013</i>
Scholarship for graduate studies, CoNaCyT	<i>2008 - 2011</i>
Technical Council member, School of Higher Studies Acatlán	<i>2005 - 2006</i>

PUBLICATIONS

Publications

Printing Orientation Optimization by Analyzing Connected Components

Garcia, J., Benes, B.

Submitted to Additive Manufacturing, June 2017.

Connected fermat spirals for layered fabrication.

Zhao, H., Gu, F., Huang, Q., Garcia, J., Chen, Y., Tu, C., Benes, B., Zhang, H., Cohen-Or, D., Chen, B.

ACM Transactions on Graphics (SIGGRAPH), July 2016.

Learning Geometric Graph Grammars

Fiser, M., Benes, B., Garcia-Galicia, J., Abdul-Massih, M., Aliaga, D.G., and Krs, V.

SCCG '16 Proceedings of the 32nd Spring Conference on Computer Graphics, April 2016.

Yturralde: impossible figure generator.

García Bravo, E., García, J.

SIGGRAPH '15 ACM SIGGRAPH Art Papers.

PackMerger: A 3D Print Volume Optimizer.

Vanek, J., Garcia, J., Benes, B., Mech, R., Carr, N., Stava, O., and Miller, G. Computer Graphics Forum, September 2014.

Clever Support: Efficient Support Structure Generation for Digital Fabrication.

Vanek, J., Garcia, J. and Benes, B.

Computer Graphics Forum, August 2014.