OPTIMAL PART ORIENTATION FOR 3D PRINTING WITH FUSED FILAMENT FABRICATION

SAMEER PURI

3D printing has made fabricating arbitrary three-dimensional parts more attainable. Fused filament fabrication (FFF) has emerged as the most popular method for hobbyist use cases, as other higher quality methods like selective laser sintering (SLS) are orders of magnitude more expensive[1]. The idea behind FFF is simple: the desired material, in filament form, is pushed through a hot end and extruded onto a build plate. Motors control axial movement and extrusion speed, building parts layer by layer. In FFF, there are many problems that can lead to poor quality prints. While most have a hardware cause, some come from user choices.

The stereo-lithography (STL) file format is widely-used to approximate parts as polyhedrons with triangular faces[2]. To print a part, its STL file is converted by a slicing software into numerical control instructions, allowing the printer to build the part layer by layer. Though there are many slicing software programs, they all have the basic functionality of orienting a part by translation and rotation[3].

Part rotation affects the amount of overhang: bottom faces of a part that are unsupported by previous layers. As shown in Figure 5, decreasing the angle between the build plate and a surface reduces print quality drastically. Slicing software often creates support structures to prevent low-angle overhangs from caving in, but quality is still poor.

Selecting the best orientation of a print should improve print quality regardless of the FFF printer and slicing software used. Finding it is an optimization problem to minimize overhang surface area by rotating the part and has been investigated in previous work[4]. In this project, I intend to write a Python 3 program that takes an STL file as input and outputs the x-axis and y-axis rotations that best orient it. While libraries will be used to read STL files and optimize mathematical operations (i.e. NumPy), I will implement the core functionality. Overhang surfaces at an angle will be computed using 3D rotation and 2D projection of faces onto the build plate plane. The objective function appears to be continuous, although further analysis is needed to confirm this – there may be some shapes for which the limits of numerical precision result in jittery function values. It however is not unimodal; consider orienting a tetrahedron: each face laid flat on the build plate is a global minimizer of overhang surface area. Computing the derivative of the objective function would be a complex, model-dependent task, so a non-derivative optimization method is needed. Multilevel coordinate search, which works well in constrained low dimensionality problems[5], is a good candidate for this problem.

References

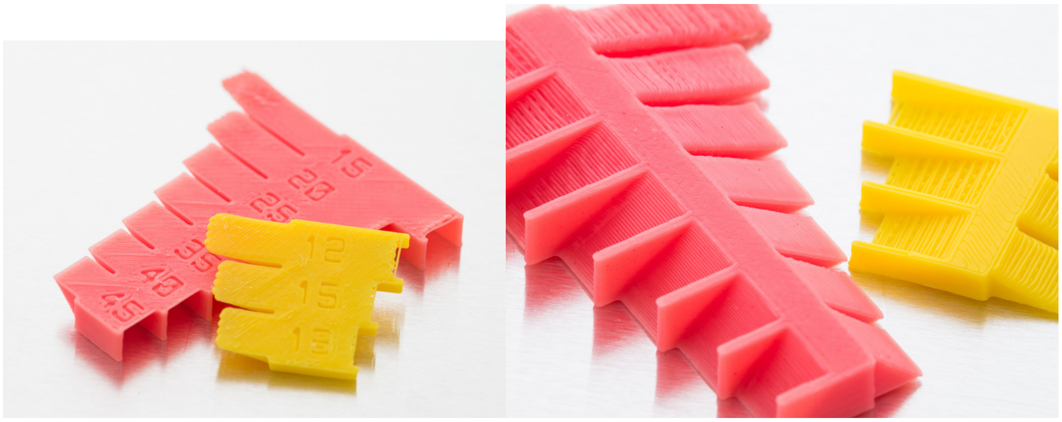
[1] MANUFACTUR3D. *A Comprehensive List of All 3D Printing Technologies.* https://manufactur3dmag.com/comprehensive-list-all-3d-printing-technologies, November 2018.

[2] All3Dp. *STL File Format (3D Printing) – Simply Explained.* https://all3dp.com/what-is-stl-file-format-extension-3d-printing/, February 2019.

[3] Jakk. *What is Slicing Software, and what does it do?.* https://www.goprint3d.co.uk/blog/what-is-slicing-software-and-what-does-it-do/, July 2016.

[4] Christoph Schranz. Tweaker - Auto Rotation Module for FDM 3D Printing. https://www.researchgate.net/publication/311765131_Tweaker_-_Auto_Rotation_Module_for_FDM_3D_Printing, December 2016.

[5] Huyer, W., Neumaier, H. *Global Optimization by Multilevel Coordinate Search.* In *Journal of Global Optimization* (June 1999), Springer, pp. 331-355. DOI 10.1023/A:1008382309369.



(A) Prints face-up with angles from build plate denoted.

(B) Prints face-down showing poor quality at lower angles.

Figure 0.1. Overhang test prints