Infill Pattern and Density Optimization for 3D Printing with Analytical Physics;
A novel approach to provide fused deposition modeling printers software to create stronger, lighter and faster to print parts.

Abstract

This project intends to optimize infill patterns and designs of 3D printed objects to provide the highest ratio of force withstood to the mass of the 3D printed object through the means of different algorithms. The first algorithm is area based optimization where the part's infill density changes depending on the relative area of a section of an area compared to the entire layer. The second is simulation based infill where the part's infill depends will be determined by simulation data. The final algorithm is a generative support structure algorithm where a part's infill design is generated as a result of the part's shape and its points of high stress.

The algorithms were tested by printing test parts in polylactic acid (PLA) 1.75mm filament through a 0.4mm nozzle. Parts created performed three different tests: top down compression, torquing, and tension. The area based algorithm had mixed results that greatly varied on the design of the shape and the precision of the voxel meshing value. As this value approaches zero, the average infill density of the part reaches it's optimized value. However, the areas of higher density do not always relate to areas that go under more stress when tested and sometimes may not contribute to the structural integrity of the object. The results of the simulation based algorithm were much more repeatable and only minutely varied based on the precision that the calculations were performed with. The generative support structure algorithm was theoretically tested through calculations that determined its possible loads and computation complexity when creating an object. The current implementation of the algorithm makes it the most optimal for object strength to mass ratio, however, the calculation process is very computationally heavy.

Further research in this topic would involve the incorporation of calculating the flexibility of certain materials. These types of calculations allow for the algorithms to compute the amount of movement, if any, when an object is under load. There are also other methods of a generative support algorithm that could lower its time complexity during the creation. Other possible solutions are to incorporate deep learning with physics to further investigations in the field and create a faster and more in-depth algorithm.

1. Introduction

1.1. Importance of 3D Printing

Three dimensional printing, a prominent form of additive manufacturing, has risen in popularity due to its precision and capabilities to create complex parts quickly. 3D printing also enables smaller companies to produce basic manufactured parts at a low cost. The lightness of 3D printed parts makes it one of the best options where light complex parts are needed. An example of this would be a rocket, where on average the cost per kilogram of a rocketship is $18,500. Creating methods to save time in printing, weight of parts, and time in designing special parts can help a wide array of research fields and has many applications.

1.2. Current Software Capabilities

Current 3D slicing softwares, such as Cura and Slic3r have only certain capabilities when it comes to multi-infill density/pattern prints. With the example of Cura, the software only allows the user to change the infill density and pattern by themselves, however, there is no automation of the system. However, researchers at the School of Mechanical Engineering in the Southeast University, China, have developed a similar algorithm that changes the infill density in order to help improve the structural integrity of a part. Their system only changes the infill on the basis of area and does not work off of any simulation data.
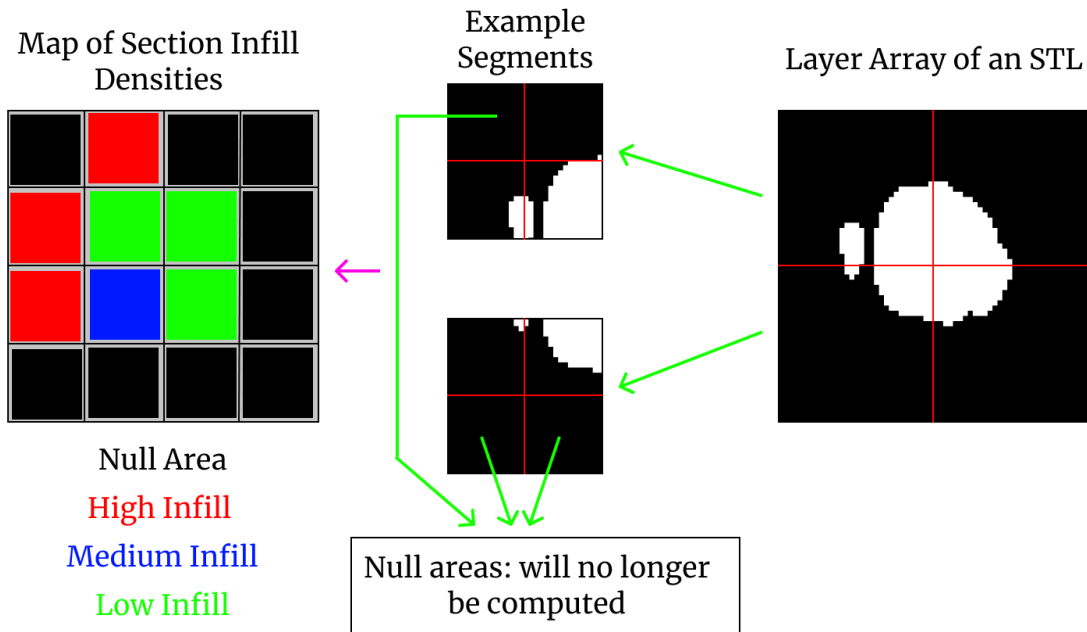
2. Engineering Goal

The goal of the project is to create an algorithm that will determine variable density and generate the pattern of infill in a 3D printed part in order to maximize the force withstood by the object and minimize the mass of the object.

3. Area Based Infill Optimization

3.1 Algorithm Design

This algorithm uses the relative area of a section of a slice in order to determine the infill of that area. Each layer is split up and each section is judged. Depending on its value, the section will be split up more or will no longer be split up. The smallest segment possible is described as the Layer Mesh Density Percentage. The example below has a LMDP value of 25

Map of Section Infill Densities

Example Segments

Layer Array of an STL

Null Area
High Infill
Medium Infill
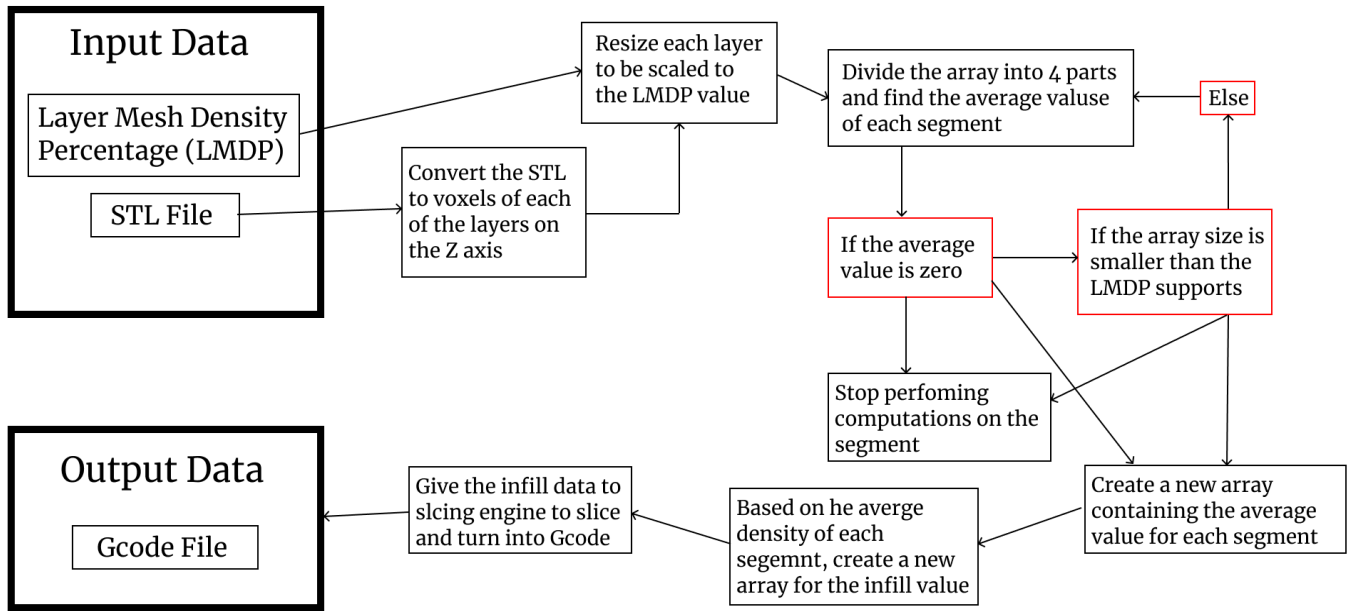Low Infill

Null areas: will no longer be computed

- Layer Mesh Density Percentage(LMDP): Determines the size of the smallest possible voxel segments made
- STL File

Process:

- The STL file will be sliced along its Z axis to create an array for each of the layers
- For each of the layers:
  - Resize the array to be large enough to be divided by the LMDP
  - Split the array into 4 separate segments:
    - If the average value of the segment is zero OR the segment size is the smallest possible
      - Stop computing that array
    - Otherwise, split each segment by 4 again
- All of the average values will be stored in a scoring array
- These will be converted in to a ranking array:
  - The higher the ranking, the higher the infill density
- Parse the infill ranking array through a slicing engine to convert to Gcode

A flow chart of this process can be seen below:

Input Data

Layer Mesh Density Percentage (LMDP)

STL File

Resize each layer to be scaled to the LMDP value

Convert the STL to voxels of each of the layers on the Z axis

Divide the array into 4 parts and find the average valuse of each segment

Else

If the average value is zero

If the array size is smaller than the LMDP supports

Stop perfoming computations on the segment

Create a new array containing the average value for each segment

Based on he averge density of each segemnt, create a new array for the infill value

Give the infill data to slcing engine to slice and turn into Gcode

Output Data

Gcode File

3.2 Data

In order to collect the data, the parts were printed on a Creality CR 10 printer and printed the three testping objects:
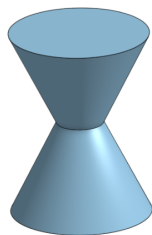
"Hourglass"
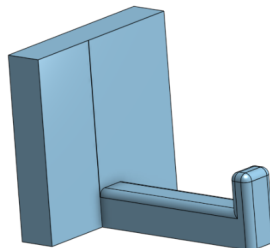
"Hook"

"Handcuffs"

Each of these objects were tested multiple times with the exception of the Hourglass shape due to long print times and lack of material. To test each of the objects, a mount was attached either to its side or on top of it and then weight were stacked on top and the performance of the object was tracked.

The data of the area based algorithm is shown below along with other baseline tests are shown below:

Top-Down Compression				
	20% Infill	60% Infill	100% Infill	Area Based Variable Infill
Trial 1 (N)	1025.8	1336.56	1845.84	1746.88
Trial 2 (N)	N/A	N/A	N/A	N/A

|				
Trial 3 (N)	*N/A*	*N/A*	*N/A*	*N/A*
Ave(N)	*N/A*	*N/A*	*N/A*	*N/A*
Mass (g)	57.7	169.5	277.4	211.6
Force/Mass Ratio	**17.77816291**	**7.885309735**	**6.65407354**	**8.25557656**
Torque				
	20% Infill	60% Infill	100% Infill	Area Based Variable Infill
Trial 1 (N)	10.34	18.62	21.35	17.96
Trial 2 (N)	12.44	19.01	20.33	19.67
Trial 3 (N)	12.68	17.43	20.8	19.34
Ave(N)	11.82	18.35333333	20.82666667	18.99
Mass (g)	5.3	8.2	10.3	7.4
Force/Mass Ratio	**2.230188679**	**2.238211382**	**2.022006472**	**2.566216216**
Tension				
	20% Infill	60% Infill	100% Infill	Area Based Variable Infill
Trial 1 (N)	177.25	325.86	444.68	399.65
Trial 2 (N)	174.44	326.7	423.56	402.6
Trial 3 (N)	176.74	326.02	425.78	401.35
Ave(N)	176.1433333	326.1933333	431.34	401.2
Mass (g)	3.5	8.2	13.4	10.2
Force/Mass Ratio	**50.32666667**	**39.7796748**	**32.18955224**	**39.33333333**

3.3 Data Analysis

From the data table as well as the graphs, there are some clear patterns that can be seen: As the mass of the printed object increased, so did the force it would withstand. However, one interesting fact to point out is that our algorithm did not always have the best mass to force held ratio. This is due to the fact that we used a relatively low LMDP value which does not provide a very detailed analysis of the part. Each object has a certain overall infill density that works best, as the LMDP is increased, the density given by the algorithm will get closer to the most optimal number.

This method of optimization has mixed results because its results highly depended on the shape of the object and the LMDP value. An example of this are concentric shapes, where the edges are always a high density. Although this algorithm does not directly help strengthen sections of a part under stress, it can be helped curing printing as it increases the density in what may be fragile sections.

4. Simulation Based Infill Optimization

4.1 Algorithm Design

   The design of this algorithm is to use a finite element method (FEM) simulation in order to determine the relative amount of infill necessary in a spot. A user provided 3D object will be given along with a material for the FEM calculations and a script to perform the tests on the object due to a lack of a graphical frontend. The general process is shown below:
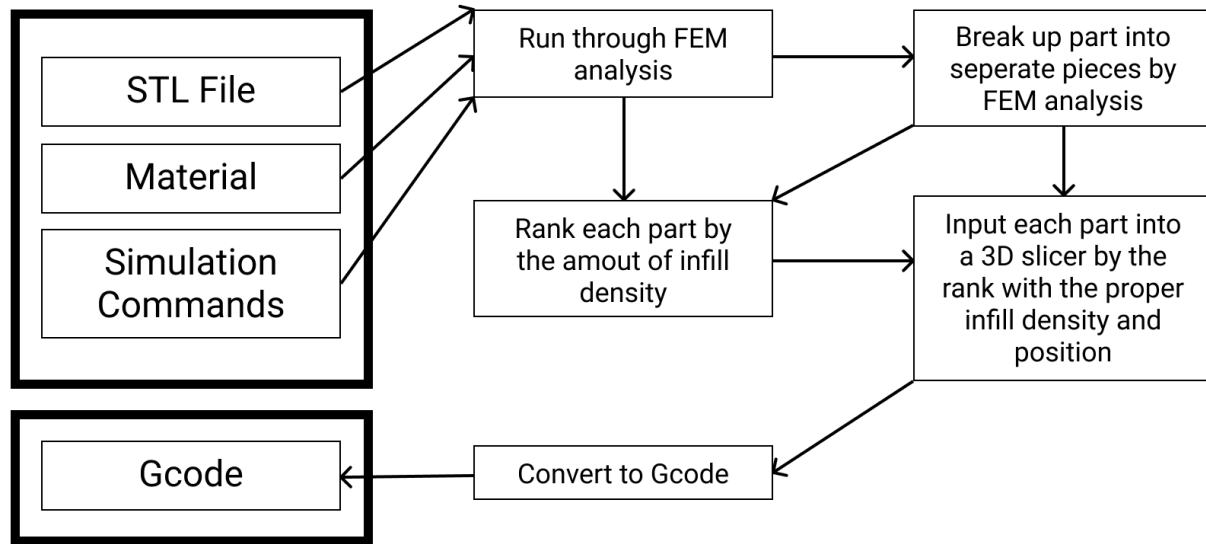
   The chart above is not very accurate and there are some missing details, primarily, the mesh size and the number of different infill densities that can exist per part. As these values increase, it becomes more computationally difficult, however, the average infill density once again approaches the most optimal value, similar to the area based optimization algorithm.

4.2 Data

   These tests used the same setup as described before.

Top-Down Compression					
	20% Infill	60% Infill	100% Infill	Area Based Variable Infill	Simulation Based Variable Infill
Trial 1 (N)	825.8	1336.56	1845.84	1746.88	1302.64
Trial 2 (N)	*N/A*	*N/A*	*N/A*	*N/A*	*N/A*
Trial 3 (N)	*N/A*	*N/A*	*N/A*	*N/A*	*N/A*
Ave(N)	*N/A*	*N/A*	*N/A*	*N/A*	*N/A*
Mass (g)	57.7	169.5	277.4	211.6	106.4
Force/Mass Ratio	**14.31195841**	**7.885309735**	**6.65407354**	**8.25557656**	**12.24285714**

Torque				Area Based Variable Infill	Simulation Based Variable Infill
	20% Infill	60% Infill	100% Infill		
Trial 1 (N)	10.34	18.62	21.35	17.96	17.79
Trial 2 (N)	12.44	19.01	20.33	19.67	18.24
Trial 3 (N)	12.68	17.43	20.8	19.34	17.94
Ave(N)	11.82	18.35333333	20.82666667	18.99	17.99
Mass (g)	5.3	8.2	10.3	7.4	6.2
Force/Mass Ratio	**2.230188679**	**2.238211382**	**2.022006472**	**2.566216216**	**2.901612903**
Tension				Area Based Variable Infill	Simulation Based Variable Infill
	20% Infill	60% Infill	100% Infill		
Trial 1 (N)	177.25	325.86	444.68	399.65	305.76
Trial 2 (N)	174.44	326.7	423.56	402.6	306.47
Trial 3 (N)	176.74	326.02	425.78	401.35	305.56
Ave(N)	176.1433333	326.1933333	431.34	401.2	305.93
Mass (g)	3.5	8.2	13.4	10.2	9.1
Force/Mass Ratio	**50.32666667**	**39.7796748**	**32.18955224**	**39.33333333**	**33.61868132**

4.3 Data Analysis

  This type of algorithm consistently did better than the other other scenarios. As seen from above, it is consistently the best algorithm and received the best results. Due to the possible errors during the testing, these values could have been even higher.

5. Generative Support Structures

5.1 Algorithm Design and Use

  This algorithm takes the approach of creating the infill in a generative method, that is, where support structures inside the part are computationally generated. This differs from current destructive techniques, which use simulations to find redundant material in a pre designed part. A generative approach allows the part to be designed itself as long as the algorithm is given constraints as to how to create the support material.

  There are three parameters for this algorithm, the file, max polynomial degree value, and beam number. The max polynomial degree value is the highest order a beam can be, as this value gets higher,

the more curves a beam can have. The beam number is the number of beams that are inside an object and make up its inner structure.

The design of this algorithm is to treat miniature structural beams as mathematical functions. (As seen on the right) The object is put through a stress analysis where the points of high stress are found and are then ranked on the amount of stress per location. Based on these points, a polynomial function is created that will go through as many points of high stress as possible.
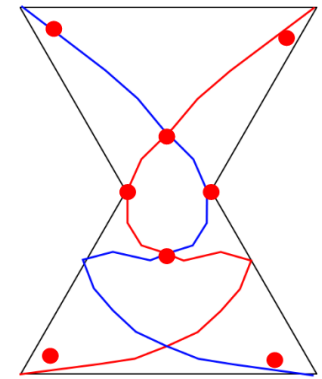
Each function is ranked based on the ranks of the points it goes through. Depending on the rank of each beam, the thickness of the beam will be defined. However, this method does not account for intersecting beams that may change the ranking of others. In order to get the best calculations, each function was split up into arches where the stress they can be determined by the Euler-Bernoulli Theorem:

Shown above is an example of two beams descibed as cubic polynomials

● Example Point of High Stress

$$\frac{d^2}{dx^2}\left(EI\frac{d^2\Delta}{dx^2}\right) = w$$

The integral of this equation will determine the stress per arch of a beam. From that data, each of the arch segments can have their own thickness depending on the amount of stress on them.

5.2 Complexity Problem

A problem emerges when the amount of the arches starts to become too large to handle. In order to account for intersections and tangests, the polynomial functions are split up into arcs following the order to the right. When two arcs cross or are tangent to each other, they become 4 arcs and when two arcs cross over each other, they become 6 arcs. The higher the beam value is for a given object, there are going to be many more arcs to compute for making this system almost computationally unuseable.
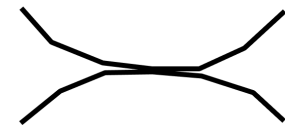

When two arcs cross over, they create 6 arcs to compute

When two arcs are tangent to each other, they create 4 arcs to compute

6. Conclusions

The work done in this project has successfully completed its task of changing infill densities based on the part and other given parameters. This allows users of this system to create optimized G Code that will take less time printing and have a better force held to mass ratio. However, there were some flaws in the methods used to determine the effectiveness of the algorithms. Due to the rudimentary methods of stacking weights above the item until rupture caused the jumps in weight help to jump up for every weight added. On top of that, the weights were not perfectly aligned atop the item and this can also decrease the possible results from a part.

7. Further Research

Continued research of this topic would involve accounting for the many other factors that involve the incorporation of the multiple other factors that affect an object under pressure. Some of these other factors are flexibility of material, effects of surrounding temperature and the effects of coatings on certain materials. There are also more optimal ways to generate support structures and the time complexity that was produced by this project was far too high and can definitely be reduced. There has also been research in combining deep-learning with physics by Princeton University as well as Google; such methods could be further solutions to make the computational process faster.

8. Bibliography

1. Achouri, Z., Amroun, N. E., & Benaissa, A. (2016). The Euler-Bernoulli beam equation with boundary dissipation of fractional derivative type. Mathematical Methods in the Applied Sciences, 40(11), 3837–3854. doi: 10.1002/mma.4267

2. Belter, J. T., & Dollar, A. M. (2014). Strengthening of 3D printed robotic parts via fill compositing. 2014 IEEE/RSJ International Conference on Intelligent Robots and Systems. doi: 10.1109/iros.2014.6942959

3. Davis, C. S., Hillgartner, K. E., Han, S. H., & Seppala, J. E. (2017). Mechanical strength of welding zones produced by polymer extrusion additive manufacturing. Additive Manufacturing, 16, 162–166. doi: 10.1016/j.addma.2017.06.006

4. Gagalowicz, A. (n.d.). Morphological modeling and deformation of 3D objects. Proceedings SMI. Shape Modeling International 2002. doi: 10.1109/smi.2002.1003546

5. Khatwani, J., & Srivastava, V. (2018). Effect of Process Parameters on Mechanical Properties of Solidified PLA Parts Fabricated by 3D Printing Process. 3D Printing and Additive Manufacturing Technologies, 95–104. doi: 10.1007/978-981-13-0305-0_9

6. Păcurar, R., Păcurar, A., Popişter, F., & Popişter, A. (2015). Finite Element Analysis to Improve the Accuracy of ABS Plastic Parts Made by Desktop 3D Printing Method. *Applied Mechanics and Materials*, *760*, 509–514. doi: 10.4028/www.scientific.net/amm.760.509

7. Pande, V., & Nataraj, P. (n.d.). Robust stabilization of Bernoulli-Euler beam by one point feedback. Proceedings of IEEE/IAS International Conference on Industrial Automation and Control. doi: 10.1109/iacc.1995.465833

8. Papacharalampopoulos, A., Bikas, H., & Stavropoulos, P. (2018). Path planning for the infill of 3D printed parts utilizing Hilbert curves. Procedia Manufacturing, 21, 757–764. doi: 10.1016/j.promfg.2018.02.181

9. Radenković, G., & Borković, A. (2018). Linear static isogeometric analysis of an arbitrarily curved spatial Bernoulli-Euler beam. doi: 10.31224/osf.io/9rgu6

10. Sansour, C., Nguyen, T. L., & Hjiaj, M. (2015). An energy-momentum method for in-plane geometrically exact Euler-Bernoulli beam dynamics. International Journal for Numerical Methods in Engineering, 102(2), 99–134. doi: 10.1002/nme.4832

11. Stava, O., Vanek, J., Benes, B., Carr, N., & Měch, R. (2012). Stress relief. ACM Transactions on Graphics, 31(4), 1–11. doi: 10.1145/2185520.2185544

12. Torres, J., Cotelo, J., Karl, J., & Gordon, A. P. (2015). Mechanical Property Optimization of FDM PLA in Shear with Multiple Objectives. *Jom*, *67*(5), 1183–1193. doi: 10.1007/s11837-015-1367-y

13. Tzes, Yurkovich, & Langer. (1989). A method for solution of the Euler-Bernoulli beam equation in flexible-link robotic systems. IEEE International Conference on Systems Engineering. doi: 10.1109/icsyse.1989.48736

14. Valencia, A. J., Nadon, F., & Payeur, P. (2019). Toward Real-Time 3D Shape Tracking of Deformable Objects for Robotic Manipulation and Shape Control. 2019 Ieee Sensors. doi: 10.1109/sensors43011.2019.8956623

15. Vosynek, P., Navrat, T., Krejbychova, A., & Palousek, D. (2018). Influence of Process Parameters of Printing on Mechanical Properties of Plastic Parts Produced by FDM 3D Printing Technology. MATEC Web of Conferences, 237, 02014. doi: 10.1051/matecconf/201823702014

16. Wang, H. (n.d.). Asymptotic analysis of eigenfrequencies of Euler-Bernoulli beam equation with structural damping. Proceedings of the 28th IEEE Conference on Decision and Control. doi: 10.1109/cdc.1989.70526

17. Wu, J., Aage, N., Westermann, R., & Sigmund, O. (2018). Infill Optimization for Additive Manufacturing—Approaching Bone-Like Porous Structures. IEEE Transactions on Visualization and Computer Graphics, 24(2), 1127–1140. doi: 10.1109/tvcg.2017.2655523

18. Zhao, Y., Chen, Y., & Zhou, Y. (2019). Novel mechanical models of tensile strength and elastic property of FDM AM PLA materials: Experimental and theoretical analyses. Materials & Design, 181, 108089. doi: 10.1016/j.matdes.2019.108089

19. Zhou, Y., Kalogerakis, E., Wang, R., & Grosse, I. R. (2016). Direct shape optimization for strengthening 3D printable objects. Computer Graphics Forum, 35(7), 333–342. doi: 10.1111/cgf.13030