AN INTEGRATED DESIGN APPROACH FOR INFILL PATTERNING OF FUSED DEPOSITION MODELING AND ITS APPLICATION TO AN AIRFOIL

Seokpum Kim[1,4], Xiang Chen[2], Gregory Dreifus[3], John Lindahl[1], Inseung Kang[4], Jung-Hyun Kim[5], Mohamed Selim[6], David Nuttall[1], Andrew Messing[1], Andrzej Nycz[1], Robert Minneci[1], Kenneth Stephenson[7], John C. Bowers[2], Brittany Braswell[2], Byron Pipes[8], Ahmed Arabi Hassen[1], Vlastimil Kunc[1,9]

[1]Manufacturing Demonstration Facility, Oak Ridge National Laboratory, Oak Ridge, TN 37831
[2]Department of Computer Science, James Madison University, Harrisonburg, VA 22807
[3]Department of Mechanical Engineering, Massachusetts Institute of Technology, Cambridge, MA 02139
[4]School of Mechanical Engineering, Georgia Institute of Technology, Atlanta, GA 30332
[5]School of Aerospace Engineering, Georgia Institute of Technology, Atlanta, GA 30332
[6]Department of Materials Science and Engineering, University of Alabama at Birmingham, Birmingham, AL 35249
[7]Department of Mathematics, University of Tennessee Knoxville, Knoxville, TN 37996
[8]School of Aeronautics and Astronautics, Purdue University, West Lafayette, IN 47907
[9]School of Materials Engineering, Purdue University, West Lafayette, IN 47907

ABSTRACT

We present a new approach to incorporate an internal stress distribution into the design of infill via fused deposition modeling of additive manufacturing (AM). This design approach differs from topology optimization, since the topology optimization of AM focuses on changing the overall shape of the product, whereas the approach we propose in this research focuses on the porous infill and remains the overall shape of the product intact. The approach presented here is effective if the overall shape is an important functioning aspect of a product and the stress is applied to the entire body, not to given localized points. As an application, we demonstrate an airfoil with its infill densities optimized based on the pressure applied during operation. Specifically, the stress of an airfoil is analyzed with operational loading conditions. The local density of the infill pattern is determined based on the computational stress analysis. The infill geometry is mathematically generated using a circle packing algorithm. Test results show that the airfoil with the optimized infill outperforms the same shape with the traditional uniform infill pattern of an airfoil having the same weight.

1. INTRODUCTION

Additive manufacturing (also known as 3D printing) is the process of building a product by joining its cross sections layer by layer. There are different AM process methods that have been developed

to process a wide array of materials such as metals, alloys, polymers, and polymer composites [1, 2]. Fused Deposition Modeling (FDM) is one of the most common AM techniques for polymer and polymer composites [3]. This technique utilizes a heated nozzle for melting and extruding polymer/composite filaments, and depositing the material on a heated platform building up the structure from bottom to the top layer by layer [1, 2]. The tool path for the extrusion head, printing speed, and the throughput (i.e. printing sequence and procedure) are identified using slicing software that generates Gcode. The quality of the printed part, and the mechanical reliability of the part are highly affected by the quality of the generated Gcode and the slicing software. Oak Ridge National Laboratory (ORNL) has developed a slicing software that accepts CAD file in STL format and generates Gcode [4].

At the FDM process, to achieve an accurate part dimensions and smooth surface quality, a slicer determines the printing procedure. Typically, the perimeter boundary of the design on a layer is deposited first, and the interior area is filled next. The interior part of the product can be partially filled to save manufacturing time, cost, and the material, and to reduce the weight of the product. However, the trade-off of choosing the partially filled interior as compared to choosing the fully filled interior is the weakening of mechanical stiffness and strength. Products with high percentage of infill amount would provide better mechanical strength and stiffness than those with low percentage of infill. Maximizing the mechanical reliability of a product with a limited amount of weight is highly dependent on the infill structure. There are several different types of infill patterns such as honeycomb patterns, plaid cross lines, and zigzag lines. The effect of the infill patterns on the mechanical response has been investigated [5]. Most slicers commonly available in the market provide features such that users can specify the percentage of the overall infill amount and the infill shape. The automated infill structure generation can be dependent on the geometry of the product so that the infill patterns are refined near the perimeter surfaces. However, the slicing and tool path planning has not accounted for the internal mechanical stress induce by the operation loads.

A computational structure analysis of a product provides the information of the locations where high mechanical stress is generated. Based on the stress profile, the locations where more material deposition is required can be determined. One example of this approach is topology optimization in which the portions with low internal stress are cut out and only the portions with high internal stress remain at the design stage. This approach has gained significant attention [6] and commercial software packages with such feature are available in the market already. This approach is especially beneficial if the applied loads are localized at given locations and the distinct stream lines of stress can be obtained. However, applications of this approach are limited if the product operates under distributed loads and the product requires infill patterns for mechanical resistance throughout the entire body. In such cases, an ideal approach is to restructure the infill patterns based on the internal stress generated during its operation.

We demonstrate an airfoil with its infill densities optimized based on a pressure load applied during operation. Specifically, the stress of an airfoil is computationally analyzed. The local density of the infill pattern is determined based on the stress profile. The infill geometry is mathematically generated using a circle packing algorithm. The infill pattern based on the internal stress profile is printed. Static load test is performed for the infill pattern from the proposed method.

2. METHODOLOGY: GENERATION OF INFILL PATTERN

2.1 Implementation of Circle Packing Algorithm

We utilize the method of circle packing generation to create hexagonal infill pattern [7-10]. This method allows circles to be refined, their centers to be connected via triangulation, a hexagonal mesh to be generated via the dual graph. In this work, we define a circle packing as follows: Let $\mathbb{T}$ be a triangulation of some compact surface $\mathbb{S}$. Each vertex in the triangulation correspondents to a circle center such that two vertices in $\mathbb{T}$ are connected by an edge if two circles are tangent to each other. Circle packing $\mathbb{P}$ realizes the triangulation $\mathbb{T}$. Circle packings are powerful and interesting mathematical tools for many reasons, but in our application, this method provides consistent and reliable configurations because circle packings are unique for a given surface (up to Möbius transform with the 2-sphere). As such, if $\mathbb{T}$ is a triangular of a disk such that for each boundary vertex v in $\mathbb{T}$ and some real-valued function $R: V(\mathbb{T}) \rightarrow \mathbb{R}^+$, then there exists a unique circle packing $\mathbb{P}$ realizing $\mathbb{T}$ where the radius of the circle in $\mathbb{P}$ corresponding to v has radius $R(v)$. These boundary radii define the aforementioned constraints of the circle packing; the radii of the boundary circles uniquely define the packing.

The rigidity of the circle packing provides both advantages and disadvantages for our application. On the one hand, the rigidity allows implementing a refinement on $\mathbb{P}$ that maintains the original constraint of the face angle sum of the boundary vertices in the triangulation. An angle sum is the total angle $\alpha(v)$ around a vertex v, and the constraints on the packing defined by the radii of the boundary vertices can also be formulated by the face angle sums of the boundary vertices. It is possible to change the radii of a subset of the circles in the packing without altering the constraints of the face angle sums along the boundary. However, this rigidity can also lead to unexpected changes in the combinatorics of the packing by slight changes in the boundary conditions imposed. As such, we further utilize the heuristic circle packing algorithm defined in [7, 8] to achieve refinement based on the simulated stress values. The algorithm is defined as follows: Given a vertex v in our triangulation $\mathbb{T}$, let N_v be the set of vertices neighboring v. Let $p(v)$ and $p(v')$ be the locations of v and v' in $\mathbb{R}^2$, and let the radius of the circle whose center is v be labeled $R(v)$. Let $U_{v'} = p(v) - p(v')$ and $\|U_{v'}\|$ be its length. Then to achieve packing within a given polygon shape we implement the following algorithm until a threshold on the error achieved on the inversive distance (defined below) is achieved:

$$p(v) := p(v) + \sum_{v' \in N_v} \frac{U_{v'}\left(\|U_{v'}\| - R(v') - R(v)\right)}{\|U_{v'}\|} \tag{1}$$

$$R(v) := \frac{1}{|N_v|} \sum_{v' \in N_v} \left(\|U_{v'}\| - R(v')\right) \tag{2}$$

We must also include a special update procedure for the boundary case, where q_v the point on our boundary polygon nearest the vertex v and $U = q_v - p(v)$:

$$p(v) := p(v) + \frac{U(\|U\| - R(v))}{\|U\|} + \sum_{v' \in N_v} \frac{U_{v'}\left(\left\|U_{v'}\right\| - R(v') - R(v)\right)}{\left\|U_{v'}\right\|} \tag{3}$$

$$R(v) := \frac{1}{|N_v| + 1}\left(\|U\| + \sum_{v' \in N_v}\left(\left\|U_{v'}\right\| - R(v')\right)\right) \tag{4}$$

The inversive distance $Inv(v,w) = \frac{\|p(v) - p(w)\|^2 - R(v) - R(w)}{2R(v)R(w)}$ measures the overlap between circles centered at $p(v)$ and $p(w)$ for vertices v and w in $\mathbb{T}$ for a circle packing $\mathbb{P}$. If the inversive distance is 1, then two circles are tangent. If the inversive distance is greater than 1, then the circles are entirely disjoint from one another. If the inversive distance is less than 1, then the circles overlap. The error that we can bound to terminate the algorithm is $\varepsilon(v,w) = 1 - Inv(v,w)$.

2.2 Mesh generation based on the Circle Packing Algorithm

Two different methods are used in the mesh generation of this work. Stephenson's algorithm calculates the unique circle packing that corresponds with a specific set of combinatorics and boundary conditions [7-9]. Bowers' algorithm calculates a flexible circle packing using relaxed constraints on specific circle tangencies in a force simulation algorithm [10].

Stephenson's method consists of two consecutive procedures [9]. The input to the algorithm is a planar graph that encodes the combinatorics and a specified set of radii for all vertices on the boundary of the graph which remain constant throughout the algorithm. In the first procedure, the packing radii for all circles are determined. Then, in the second procedure, the circles are laid out tangent to the previous circles. It should be noted that since the circle packing is uniquely determined in part by the combinatorics, the combinatorics must be altered in order for the generated mesh to be refined. A creative but effective way is to increase the number of vertices in select regions of a graph by inserting new vertices into triangular faces in those regions and essentially partitioning the faces into smaller faces with some edge reconnections. The increase in the number of vertices and corresponding connecting edges lead to new combinatorics that determined denser circle packings.

Bowers' method [10] is also iterative, however the circle radii and locations are determined simultaneously. The input is a planar graph and specified minimum and maximum inversive distances allowed between circles. All circles get assigned an arbitrary initial radius and are placed in an initial grid configuration. For each circle associated, calculate the forces acting on the circle's location and radius from neighboring circles. This is done through the use of inversive distances between two circles. If two neighboring circles are overlapping, then there is a force repelling them apart location-wise and there is a compressing force on the radius to reduce the overlap. On the other hand, if two neighboring circles are disjoint, then there is a force pulling them together location-wise and there is an expanding force on the radius to close the gap. Once the forces are calculated, the forces are applied to the circle's location and radius. This process continues until all neighboring circles in the graph are tangent or have their inversive distances within the limited range specified by the minimum and maximum inversive distances.

Both circle packing algorithms are implemented using C++ to allow for a smoother integration between the circle packing software and already existing slicer software for 3D printing. The C++ implementation is designed as a library of functions that create and modify graph data structures into circle packings using various user defined parameters.

2.3 Incorporation of Mechanical Stress

Building on top of the circle packing implementation, mechanical stress data was incorporated as a way of using physical data to custom refine the meshes generated through circle packings. The stress data was represented as a planar grid with a stress value associated with each cell in the grid. This grid was projected on the graph data structure and the stress value for each cell determined the stress value for the corresponding region of the graph that was in the same location. These stress values were used in two ways. In the first way, the stress values were used to select specific parts of the graph that were located in a region with stress greater than a user defined threshold. This method allowed for the generation of meshes that were denser in high stress regions. In the other way, the stress values were used as additional forces on the radii of circles over a continuum with high stress corresponding to compression and low stress corresponding to expansion. This method allowed for a more gradual transition in the density of the mesh from high stress to low stress regions.

3. CASE STUDY: AIRFOIL FOR 3D PRINTED DRONES

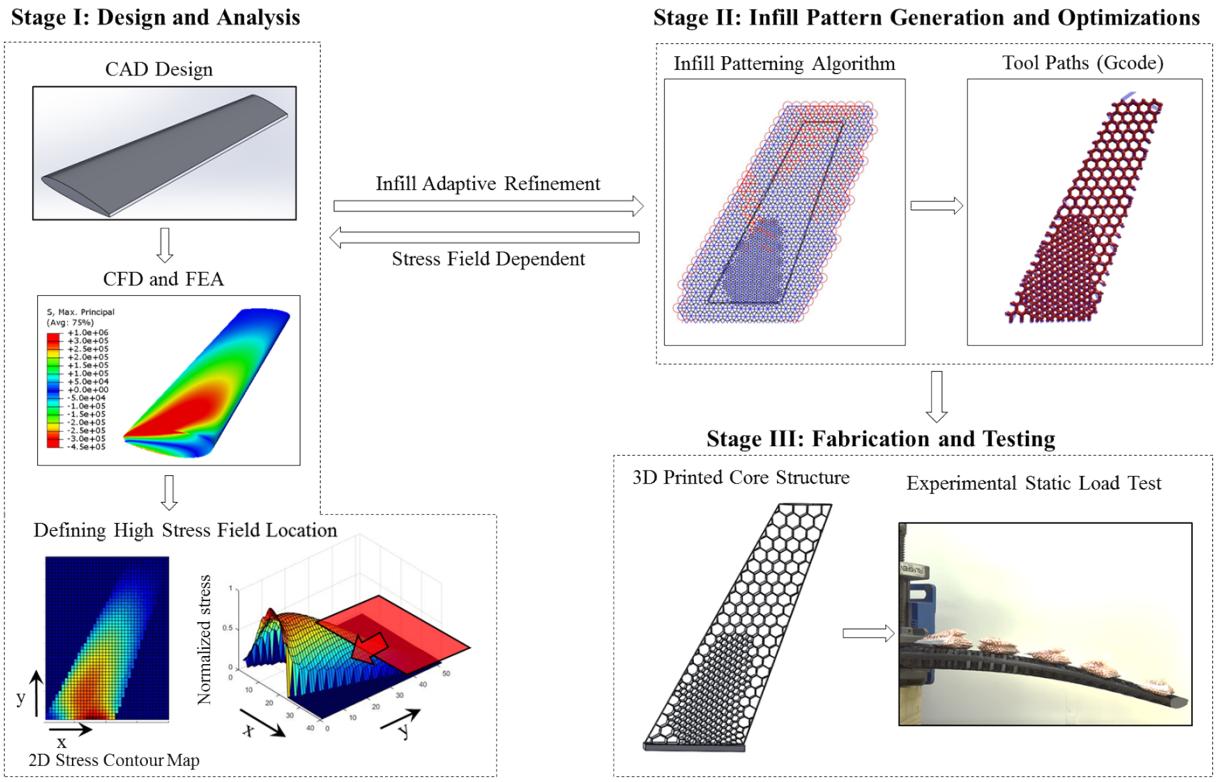

Figure 1: Integrated design approach for the infill patterning of an airfoil based on internal stress

In this study, we propose a design methodology that accounts for optimization of the infill pattern based on the stress profile from computational structure analysis. Airfoil wing for a 3D printed drone is the investigated case study in this work. The approach consists of several steps shown in Figure 1. The process involves three different stages. Stage I: the shape of the desired part is designed and its target weight is determined. Then, a computational analysis is performed to predict the internal stress of the part. Stage II: the stress analysis results influence the local density of the infill (or the size of an individual shape) throughout the part. After Stage II, the process can go back to Stage I where the computational stress analysis is performed with the product infill structure. Once the new stress profile is obtained from the computational stress analysis, the infill density is re-calibrated and re-defined. This iteration cycle continues until the maximum local stress value settles under the threshold requirement. Although we propose an iterative feedback process between computational stress analysis and the design of infill pattern, for the application in this paper, we have performed the computational stress analysis only for the original design. The optimized infill pattern obtained is used to define the tool path in a slicer and generate Gcode. Stage III: 3D printing of the optimized infill pattern and testing of the final fabricated part.

3.1 Stage I: Design and Analysis

3.1.1 Estimation of the Lift Force

The size of an airfoil we design is similar to those available for commercial drones for a leisure use. The typical wing spans are in the range of 500 mm each side. The airfoil we designed has a span of 384 mm as shown in Figure 2. The tailing edge of the wing is cut off and not included for designing the infill pattern.

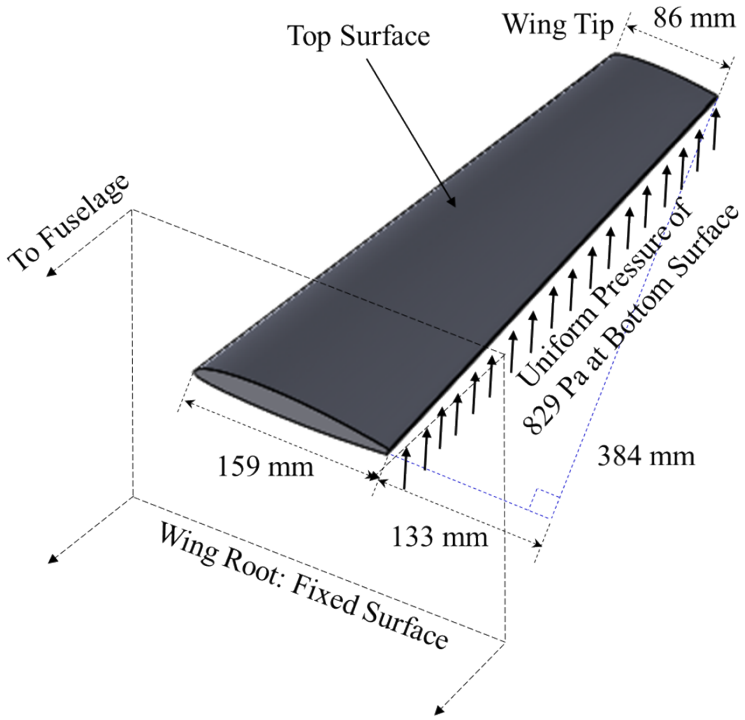

Figure 2: Schematic showing the dimensions of the wing, and the boundary and loading conditions

The results from Computational Fluid Dynamics (CFD) analysis predicted that a lift force in the range of 2 - 39 N is generated from the wing corresponding to wind speeds in the range of 9 - 36 m/s. Table 1 shows the wind speed and the corresponding drag and lift force results, assuming 20° angle of attack. The analysis was performed using ANSYS Fluent R16.2 with the air density of 1.225 kg/m^3. The maximum wind speed considered (i.e. 36 m/s) is slower than the world's record for a small scale 3D printed drone (67 m/s) [11], but much faster than commonly available drones (~10 m/s). Figure 3 shows the pressure distribution profile of the wing with a speed of 9 m/s.

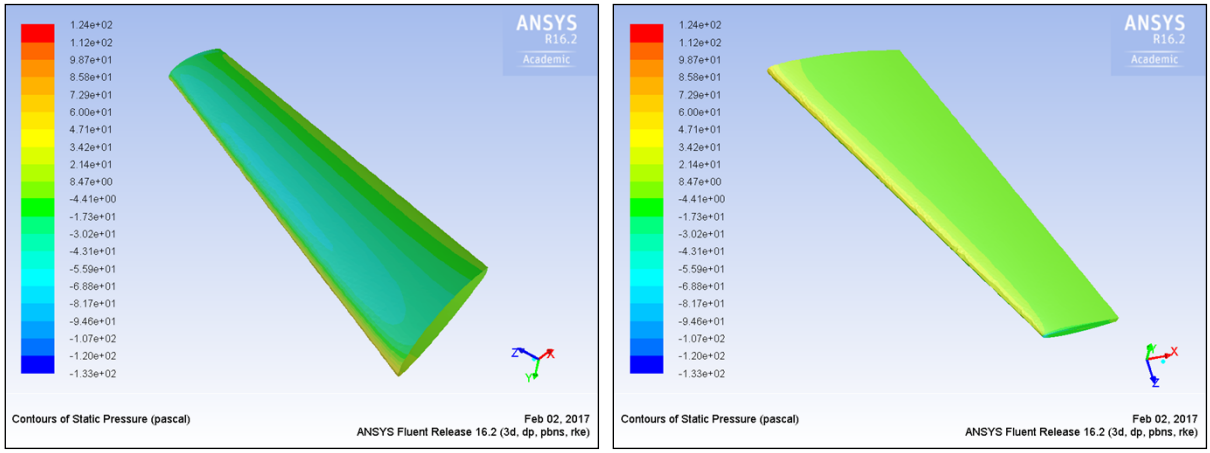

Figure 3: CFD results for pressure profile of a wing; a) Top surface of the wing, and b) Bottom surface of the wing

Table 1: Wind speed and the corresponding forces for the wing for attack angle of 20°

Speed (m/s)	Drag Force (N)	Lift Force (N)
9	0.76	2.13
18	3.17	9.35
36	13.03	39.20

3.1.2 Structural Analysis for the Internal Stress of the Airfoil

The pressure profile obtained by the CFD calculations can be used for defining a point-by-point structural load, and the experimental static load test can be performed based on the load distribution profile obtained from CFD calculations. However, to simplify the experimental setup, we use a uniform pressure on the bottom surface, and the magnitude of the pressure is estimated based on the lift force from CFD results.

Table 2: Mechanical properties of the printed infill material

Material	ABS plastic
Young's modulus (GPa)	2.14
Poisson's ratio	0.35

A commercial Finite Element Analysis (FEA) tool, ABAQUS 2016, is used for computational structural analysis. A total number of 264 k tetrahedral elements was used. To avoid shear locking,

the elements use a quadratic interpolation function. The numerical framework utilizes elastic constitutive relations. Table 2 shows the material properties of the printed infill material. The wing root is fixed, but the wing tip and the other boundary surfaces are not constrained. Uniform pressure of 829 Pa (0.12 psi) is applied to the wing bottom surface. This boundary and loading configurations represent a wing which is attached to the fuselage and under a uniform lift pressure, see Figure 2. The pressure of 829 Pa (0.12 psi) is equivalent to 39 N which is the maximum lift force considered.

The results show that the maximum internal stress is generated at the root of the wing and the internal stress gradually decreases as the distance from the wing root increases. Since we assume that the Acrylonitrile Butadiene Styrene (ABS) plastic infill would show elastic behavior with brittle fracture at its ultimate strength, we focus on the maximum principal stress profile. Since von Mises stress profile shows a similar stress distribution as compared to the profile of maximum principal stress, the choice of the stress type does not affect the refinement of the infill pattern. Both profiles are shown in Figure 4.

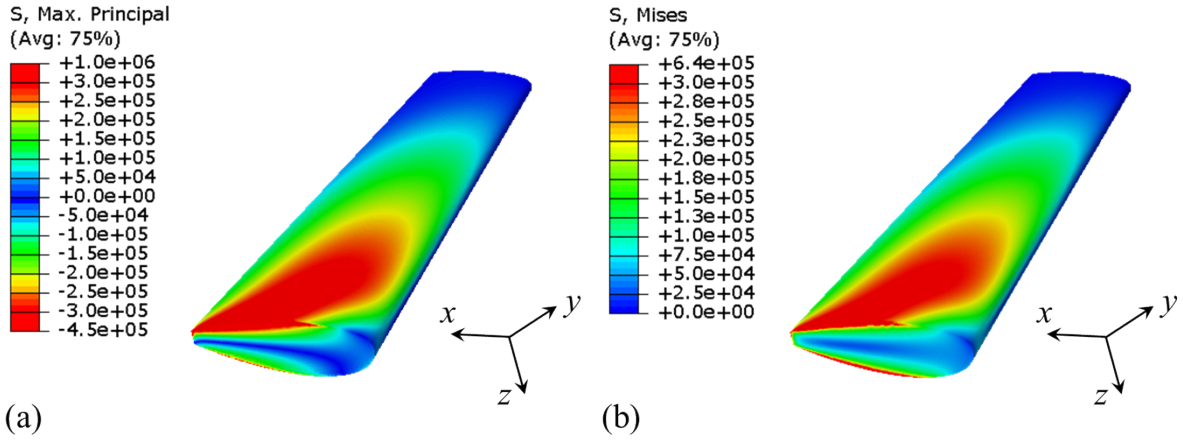

Figure 4: The distribution of internal stress in the wing; (a) Maximum principal stress, and (b) von Mises stress

3.2 Stage II: Infill Pattern Generation and Optimizations

3.2.1 Infill Patterning

The circle-packing algorithm discussed in Section 2 is implemented and we have successfully designed the infill pattern as shown in Figure 5a-b. The average circle sizes are calibrated such that the two-infill patterns have the same net volume (i.e., uniform infill with 127.7 cc and refined infill with 127.8 cc at the design stage). A Stratasys Fortus 400MC is used to print the infill designs using ABS plastic. The printing of each design takes 10 – 11 hours. The printed infills have a weight of 87.5g and 88.6g for uniform infill pattern and refined infill pattern, respectively.

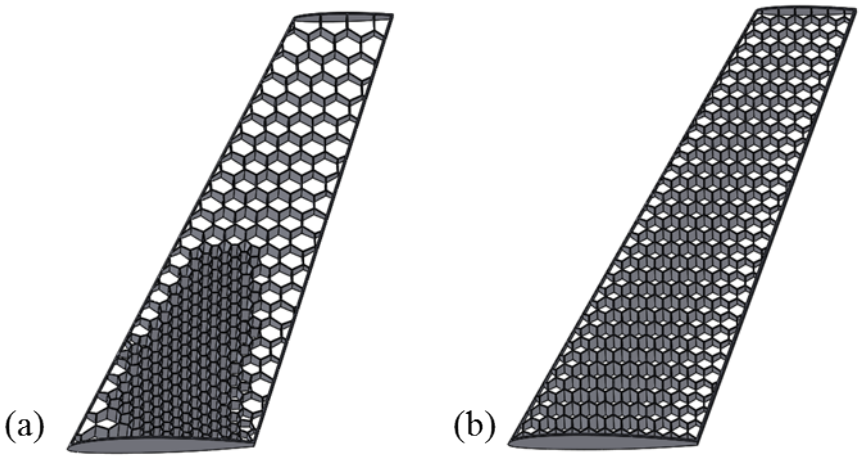

Figure 5: Airplane wings with the same weight: (a) Optimized infill pattern using circle packing algorithm (88.6 g) (b) Uniform infill pattern (87.5 g)

3.3 Stage III: Fabrication and Testing

3.3.1 Experiment: Static Load Test

Static load test is performed to evaluate the stiffness of the wings. The test is conducted to simulate the uniform pressure applied to the bottom surface of the wing. The bottom surface is divided to five equally spaced sections from the wing tip, as shown in Figure 6. Table 3, shows different load weights placed at the corresponding sections. It should be noticed that the weight added to each section is calibrated to apply a constant load per unit area for these five sections of the wing. A total of 6 experiments were conducted with a total load of 250 g, 500 g, 750 g, 1000 g, 1250 g, and 1500 g for Test 1, 2, 3, 4, 5 and 6 respectively. At each test, the corresponding deflection is measured using remote imaging and image processing techniques. A camera (Nikon D5500) is used to capture the corresponding deflection and Matlab software is used to post-process the image data. The relation between the deflection and the distributed load can be calculated based on the conventional cantilever beam theory,

$$\mathrm{w} = \delta \times (8EI)/\, l^4 \tag{5}$$

where δ is deflection, w is the load density, l is the length of the beam, E is the elastic modulus, and I is the moment of inertia. This theory provides a linear relation between the deflection and the applied load.

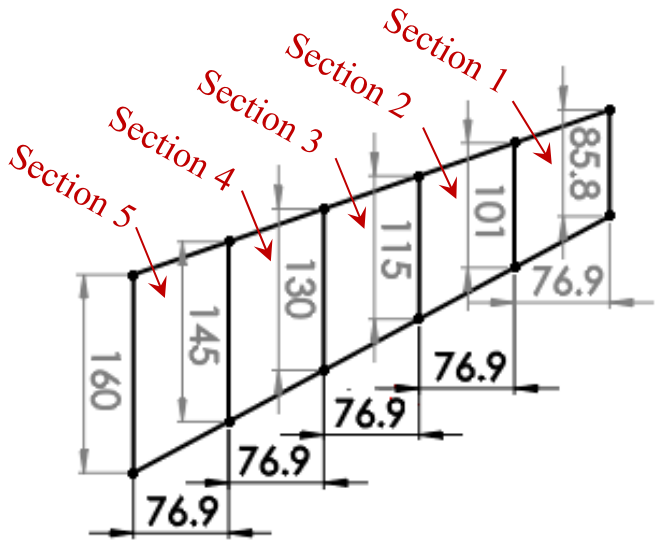

Figure 6: Equally spaced sections of the wing maesuerem from the wing tip (Units are in millimeters)

Table 3: Load weights of corresponding sections (The added weight for each section is calibrated to apply a constant load per unit area)

	Test 1	**Test 2**	**Test 3**	**Test 4**	**Test 5**	**Test 6**
Section 1 (g)	38	76	114	152	190	228
Section 2 (g)	44	88	132	176	220	264
Section 3 (g)	50	100	150	200	250	300
Section 4 (g)	56	112	168	224	280	336
Section 5 (g)	62	124	186	248	310	372
Total Load Weight (g)	**250**	**500**	**750**	**1000**	**1250**	**1500**

Figure 7 shows the photos of the wings with distributed loads. The wing on the left figure has the refined infill pattern and the wing on the right figure has the uniform infill pattern. Figure 8 shows deflections of the wings for different loading conditions (i.e., Test 1 - Test 6) with uniform infill and with optimized infill. Figure 9 shows a comparison of the defection of the wing with uniform infill and the wing with optimized infill at different loading conditions. It is noticed that the uniform pattern has higher deflection than the optimized pattern. Figure 10 shows a linear relation between the load weight and the deflection, which is expressed by the following relation,

$$w = c \times \delta \tag{6}$$

where δ is deflection, w is the load weight, c is a stiffness coefficient of the wing. The infill optimization increases the stiffness of the structure by 49.3% as reported in Table 4.

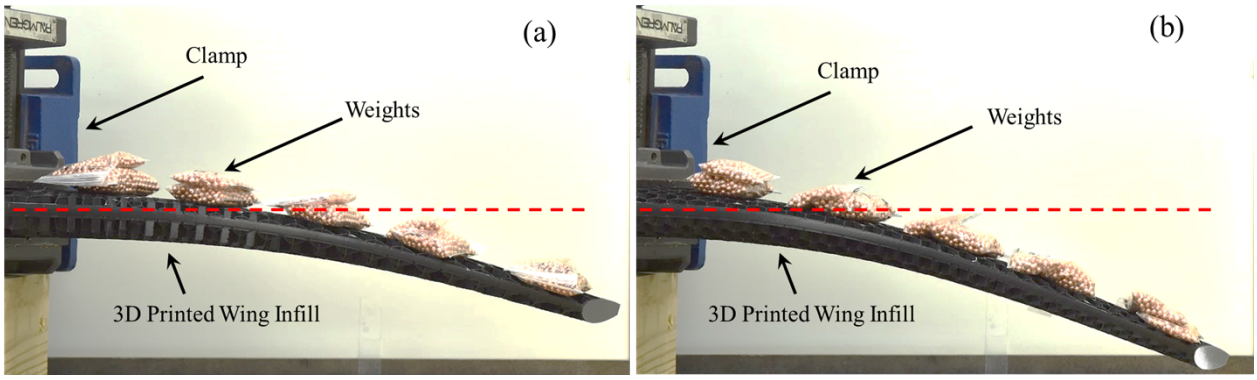

Figure 7: Static load test for the printed wing infill under the load weight of 1.0 kg; (a) Optimized infill, and (b) Uniform infill

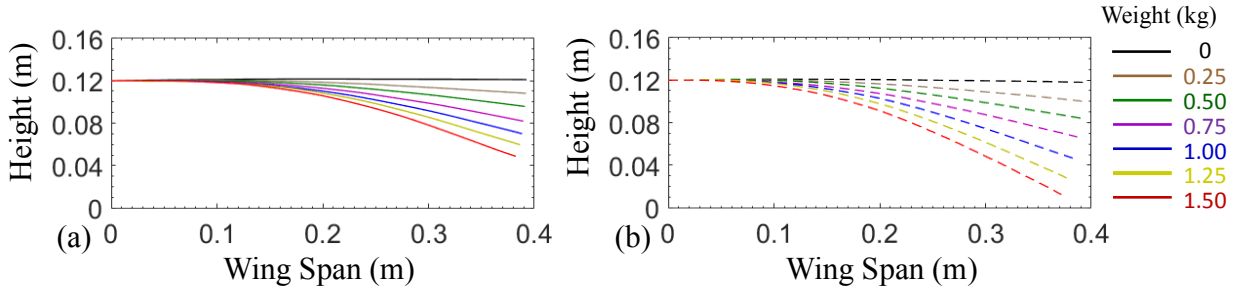

Figure 8: Measured deflection for the printed airfoil infill; (a) Optimized infill, and (b) Uniform infill

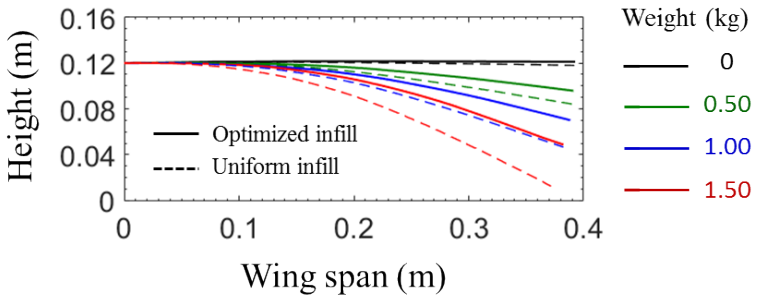

Figure 9: Comparison for measured deflection of optimized infill printed airfoil and uniform infill printed airfoil

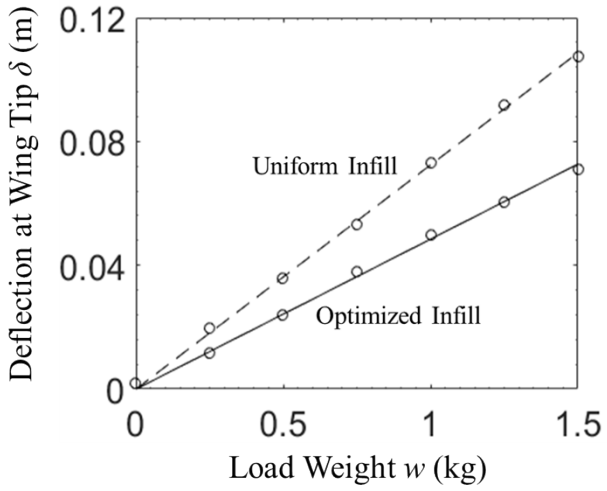

Figure 10: Deflection at the printed wing tip as a function of load weight

Table 4: Stiffness coefficient of the wings obtained from the static load test

	Uniform Infill	**Optimized Infill**
Stiffness c (kg/m)	13.8	20.6

4. CONCLUSIONS

We propose a new approach to incorporate an internal stress distribution into the design of infill via fused deposition modeling of additive manufacturing. We have successfully implemented circle packing algorithms to generate infill pattern whose size can be optimized based on the mechanical stress levels. As an application, we demonstrate an airfoil with its infill densities optimized based on a given pressure. We have performed static load tests and results show that the airfoil with the refined infill pattern outperforms the traditional uniform infill pattern of an airfoil having the same weight by ~50% in terms of stiffness.

ACKNOWLEDGEMENTS

Research sponsored by the U.S. Department of Energy, Office of Energy Efficiency and Renewable Energy, Industrial Technologies Program, under contract DE-AC05-00OR22725 with UT-Battelle, LLC.

5. REFERENCES

1. Mellor, S., L. Hao, and D. Zhang, *Additive manufacturing: A framework for implementation.* International Journal of Production Economics, 2014. **149**: p. 194-201.
2. Gao, W., et al., *The status, challenges, and future of additive manufacturing in engineering.* Computer-Aided Design, 2015. **69**: p. 65-89.

3. BRENKEN, B., et al. *Fused Deposition Modeling of Fiber-Reinforced Thermoplastic Polymers: Past Progress and Future Needs*. in *Proceedings of the American Society for Composites: Thirty-First Technical Conference*. 2016.
4. Love, L.J., *Cincinnati Big Area Additive Manufacturing (BAAM)*. 2015, Oak Ridge National Laboratory.
5. Baich, L., G. Manogharan, and H. Marie, *Study of infill print design on production cost-time of 3D printed ABS parts.* International Journal of Rapid Manufacturing, 2015. **5**(3-4): p. 308-319.
6. Tomlin, M. and J. Meyer. *Topology optimization of an additive layer manufactured (ALM) aerospace part*. in *Proceeding of the 7th Altair CAE technology conference*. 2011.
7. Dreifus, G., et al., *A new approach to tool path generation in additive manufacturing*, in *Symposium on Computational Fabrication*. 2017: Cambridge, MA.
8. Dreifus, G.D., et al., *Path Optimization Along Lattices in Additive Manufacturing Using the Chinese Postman Problem.* 3D Printing and Additive Manufacturing, 2017(submitted).
9. Collins, C.R. and K. Stephenson, *A circle packing algorithm.* Computational Geometry, 2003. **25**(3): p. 233-256.
10. Bowers, J.C. and P.L. Bowers, *Ma-Schlenker c-Octahedra in the 2-Sphere.* arXiv preprint arXiv:1607.00453, 2016.
11. Cosgrave, J., *World's fastest 3-D printed drone takes flight*, in *CNBC*. 9 Nov 2015.