International Journal of Innovative Solutions in Engineering
2025, Vol. 1 No. 1 (pages 25–32)
DOI: 10.47960/3029-3200.2025.1.1.25

Research Article

OPEN ACCESS

A Newly Modeling Procedure for Different Morphological Distributions of Unit Cells in Lattice Structure with Constant Relative Density

Mario Soldo[1]

[1]Faculty of Mechanical Engineering, Computing and Electrical Engineering, University of Mostar, Bosnia and Herzegovina

Abstract
The rapid growth of additive manufacturing has enabled the broad application of lattice structures in various engineering fields. The challenge of fitting lattice structures into the final product necessitates using different software packages. Rhinoceros's extension Grasshopper allows the incorporation of various morphological distributions of unit cells into simple and complex product designs. This work aims to design an algorithm to successfully fit unit cells on rectangular and curved surfaces in a lattice structure. In contrast, the relative density of the lattice structure remains constant. The ultimate goal of this work is to define a Grasshopper algorithm that will allow unit cells of lattice structures to fit into more complex product designs while ensuring that the material usage for the additive manufacturing of the lattice structure remains constant.

Article history
Received: 28. 12. 2024.
Revised: 10. 01. 2025.
Accepted: 12. 01. 2025.

Keywords
Unit Cell,
Lattice Structure,
Morphological Distribution,
Relative Density,
Rhinoceros,
Grasshopper Algorithm.

1 Introduction

Lattice structures have become the "holy grail" for various researchers over the last decade due to the widespread use of additive manufacturing to obtain final usable products. The application of lattice structures spans various spheres of engineering: from medicine [1], where lattice structures can be applied in femur implants, to the sports industry [2], where they can be integrated into leg guards, gloves, surfboards, helmets and more, as well as in the aerospace [3] and automotive [4] industries to reduce the weight of airplane and car parts, in the field of vibration isolation [5] and so on. Many researchers are modeling lattice structures with a rectangular morphological distribution of unit cells featuring different patterns of unit cell multiplication, as shown in the review article [6]. Recently, the use of functionally graded lattice structures has become increasingly common, where, in addition to changing the dimensions of the unit cell, the relative density of the lattice structure layers is also often modified [7 - 9]. Only a few research studies describe the modeling procedure for lattice structures whose morphological distribution of unit cells is not rectangular but follows the final shape of the product design with various morphological distributions of unit cells, known as conformal lattice structures [10 - 11]. The ability to adapt unit cells has undoubtedly been facilitated by the Rhinoceros program with the Grasshopper extension, which has been increasingly used in scientific research lately [10 - 19]. However, no works were found where the relative density was maintained as constant by changing the morphological distribution of unit cells in a lattice structure. Liu et al. [10] modeled a custom-fit helmet with different conformal lattice structures designed by considering von Mises and Principal stress. A shortcoming of this research is the relative density, which was not kept constant, and the use of the Pufferfish plugin in the Grasshopper extension. Dall Fabbro et al. [11] also use the Grasshopper extension in the Rhinoceros program to model four different conformal lattice structures in an automotive part while relative density is changeable. García-Domínguez et al. [12] include infill optimization and examine the difference between optimization results with and without minimum diameter restricted in lattice structure embedded in a cubic sample using

Contact Mario Soldo, mario.soldo@fsre.sum.ba, Faculty of Mechanical Engineering, Computing and Electrical Engineering, University of Mostar, Bosnia and Herzegovina
©2025 by the Author(s). Licensee IJISE by Faculty of Mechanical Engineering, Computing and Electrical Engineering, University of Mostar. This article is an open-access and distributed under the terms and conditions of the CC BY 4.0 (https://creativecommons.org/licenses/by/4.0/)

Grasshopper extension without caring for lattice structure relative density. Geyer and Hölzl [13] incorporate a lattice structure into a gear and bicycle seat design using the Rhinoceros program and Grasshopper extension with included relative density calculation but do not explain the lattice implementation process in detail. Also, this research contains cutting lattice structure across the product's boundary surfaces, which leads to modification of unit cell topology. Mahmoudi et al. [14] are investigating the possibilities of improving the mechanical properties of sandwich structures with homogeneous and functionally graded unit cells in lattice structures by changing the relative density using the Grasshopper extension. The main goal of this research is to design and clearly define an algorithm in the Rhinoceros program with the Grasshopper extension that will enable different morphological distributions of unit cells in a lattice structure, which will successfully achieve equal relative density through changing crucial unit cell parameters. The defined algorithm will not use external extensions or plugins outside the Grasshopper domain and will be subject to future changes.

2 Difficulties in defining unit cell parameters in Grasshopper

By starting the Grasshopper (Included in Rhinoceros, Version 7.0. Robert McNeel & Associates, Seattle, WA, USA) extension and defining a "Geometry" block, the previously created wireframe model of the Simple Cubic unit cell is selected in the Rhinoceros (Version 7.0. Robert McNeel & Associates, Seattle, WA, USA) program. A simple Cubic unit cell was chosen as one of the most straightforward representatives of the unit cell topology and as one of the most superior unit cell types for potential future investigations of compression properties [20]. By defining a "Multipipe" block and connecting the output of the previously described "Geometry" block to the first input of the "Multipipe" block called "Curves," the wireframe model is given a circular cross-section on the struts, which encompasses the entire unit cell topology (Figure 1). The "Multipipe" command has several characteristic inputs that need to be well understood to achieve complete control when

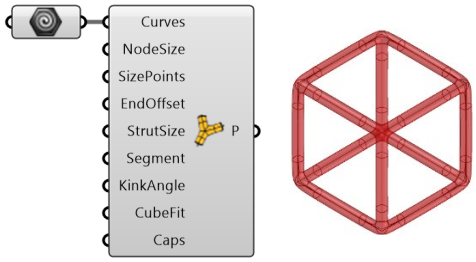

Figure 1 Connecting the "Geometry" block to the "Curves" of a "Multipipe" (left) and Simpe Cubic preview (right)

defining the morphology of the unit cell, which affects the overall relative density of the lattice structure.

The second input of the "Multipipe" block is called "NodeSize," and by connecting it to the output of the block "Slider," which determines a numerical value (Figure 2), the radius of the node – the place where the struts connect, is defined. According to the automatic settings of the Grasshopper extension, the defined node radius is set to a value of 0.50 mm. The closely related input of the "Multipipe" block, "SizePoints," can be used for more complex topologies and morphologies of unit cells with multiple nodes whose radius does not need to be defined by the same value. By defining the node points, it is possible to manipulate which node will occupy which radius. It is important to emphasize that by selecting the "NodeSize" value, the radius of the strut is directly defined. Another valuable note in the later analysis of the unit cell is the "Flatten" command within the "Multipipe" block. The "Flatten" command allows the unit cell to behave as a single unit after generating the solid model (using the "Bake" command). In this way, all supports will be connected to defined nodes and will not behave as separate units.

A significant problem can occur while defining "StrutSize" if a larger or smaller strut diameter is required because the node size remains constant (by default, the node has a radius of 0.50 mm). It is

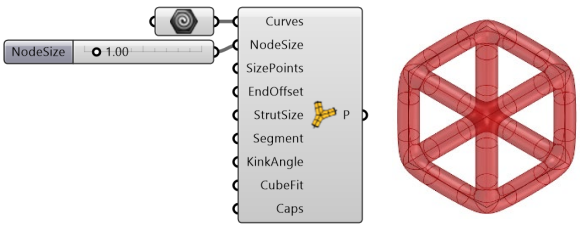

Figure 2 "NodeSize" setting by adding "Slider" block with value 1.00 (left) and Simple Cubic preview (right)

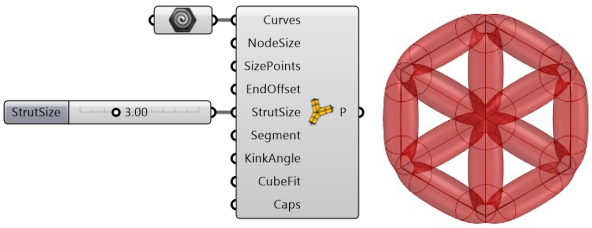

Figure 3 Defining the strut size diameter via "NodeSize" (left) and a node issue in the preview of a Simple Cubic (right)

important to note that "StrutSize" defines the strut diameter, and "NodeSize" specifies the node radius. The issue of defining larger "StrutSize" values is illustrated in Figure 3, where the node size remains determined by the program settings.

Closely related to the "StrutSize" input within the "Multipipe" block is the "EndOffset" input. It defines the distance between the first loop of the strut and the node. If a value of 0 is defined (as shown in Figure 4), there will be no distance between the first loop of the strut and the node, and thus the transition will be ideally smooth. This results in the rounding of the angle of 90° in a Simple Cubic unit cell. If a value greater than 0 is defined, the specified value will be multiplied by the node size value (Figure 5). This value defaults to 1 according to the Grasshopper program settings if it is not defined. In the research presented in this paper, the "EndOffset" input is omitted; thus, the original Grasshopper program settings are employed.

By increasing "EndOffset," a bubble appearance of the unit cell can be created, known as the bubble effect. The issue with defining unit cell parameters by combining the "NodeSize" input and the "StrutSize" input is that the "bubble effect" of the strut may be produced, making it preferable to avoid such an input definition. In addition to the defined input data within the "Multipipe" block, there is the "Segment" command, which describes the number of segments between two "EndOffsets," and according to the

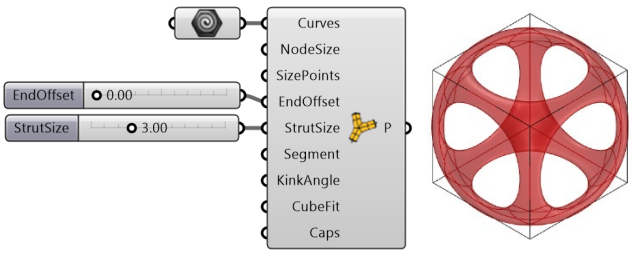

Figure 4 "EndOffset" defined with a value of 0 (left) and preview of a Simple Cubic unit cell (right)

program settings, it is 1; the "KinkAngle" command, which represents the precision of the curves on which the "Multipipe" is based; the "CubeFit" command which is used when the "Multipipe" is defined by cubic shapes to ensure that the angles are as precise as possible; and the "Caps" command. The "Caps" input is determined by the program settings with a value of 1, indicating the closure of the unit cell at the open ends. It can also be set without closure, with a value of 0, or when the end of the cell is cut off straight, the value is 2. The parameters listed will not be the subject of further study.

By examining the "Multipipe" command in detail, it can be concluded that the most significant influence on the precise definition of a Simple Cubic unit cell, for this research, comes from "Flatten" and "NodeSize." "Flatten" allows all struts to behave as compact units while connected to nodes. By defining the radius value of "NodeSize," the strut diameter value is automatically set without creating an unwanted bubble effect. These two settings are considered crucial parametric settings for defining the morphology of the unit cell.

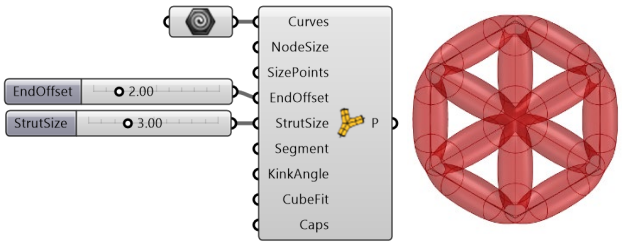

Figure 5 "EndOffset" defined with a value of 2.00 (left) with a preview of a Simple Cubic with visible bubble effect (right)

3 Methodology for generating different morphological distributions of unit cells in lattice structures

Figure 6 illustrates the methodology employed in creating the Grasshopper algorithm. The topology or unit cell type is defined in the first phase. In this research, the focus is on the Simple Cubic unit cell. In the second phase, the upper and lower surfaces are defined, which will serve as the boundary surfaces of the volume of the design. An essential enhancement of this algorithm is that the upper and lower surfaces do not necessarily have to be flat but can also be curved. In the third phase, the morphology of the

lattice structure is defined. The lattice structure's height, width, and length shape the morphological distribution of unit cells. Then, the product design volume is divided into sections. Surfaces can be flat or curved with varying distances. The heights of the unit cells are precisely defined by the distance between adjacent surfaces within the specified volume. The multiplication pattern in the x and y directions in Grasshopper determines the length and width distribution. The node's radius is established after defining the unit cells' dimensions. In this research, an algorithm was developed to automatically determine the value of the relative density by assigning a node radius value. Since relative density is closely related to material use, this approach influences total material consumption in product design production. A desired relative density of 0.2 was selected, indicating that the lattice structure's porous (hollow) part constitutes 80%. The established relative density determines the radius value, yielding an optimized lattice structure ready

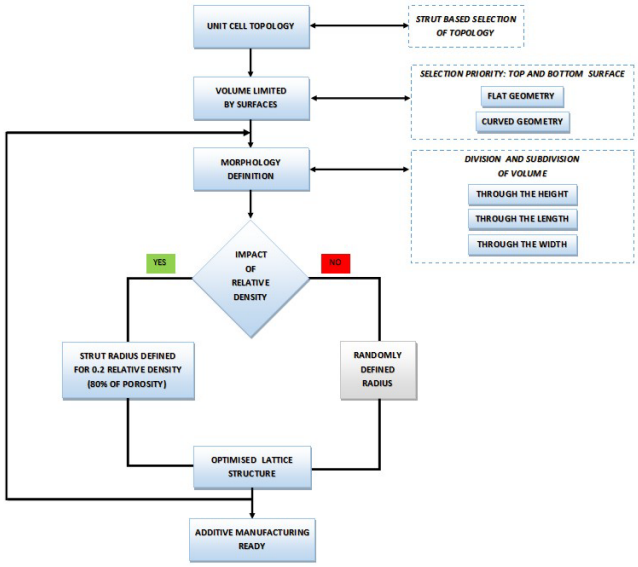

Figure 6 Grasshopper algorithm methodology

for additive manufacturing.

4 Algorithm for generating different morphological distributions of unit cells in lattice structures

In the Rhinoceros program, using the "Surface" command, the Top and Bottom surfaces are defined, representing the limits of the product design volume. The length of the side (a) of the defined square surface is 50 mm, and the distance between the square surfaces is 50 mm. These two surfaces represent the boundaries of the product design, precisely the boundaries of the cube design (Figure 7). The two surfaces are perpendicular to the Front view and parallel to the Top view in the Rhinoceros program. After defining the boundary surfaces, two different morphological distributions of unit cells are established through surfaces whose distances are specified in Figure 7. The values of the distance between the adjacent surfaces determine the height of the unit cells. The morphological distribution of unit cells according to a parabola, as shown in Figure 8 on the right, was chosen precisely to show that it is possible to distribute unit cells not only according to a rectangular spatial distribution, as shown in Figure 8 on the left, but also according to the curved geometry of the product design.

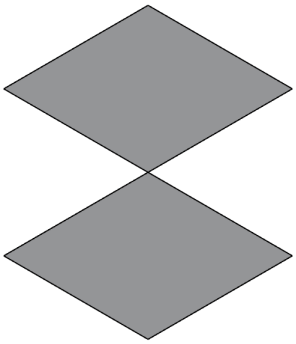

Figure 7 Defining the volume boundaries of product design across surfaces

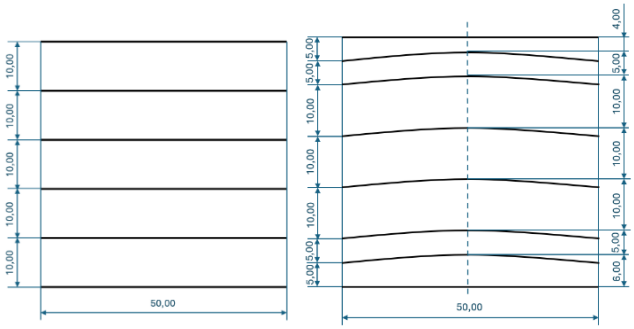

Figure 8 Dimensional characteristics of surfaces for rectangular (left) and parabolic (right) unit cell morphological distributions

In the Grasshopper extension, two "Surface" blocks are further defined and connected to the defined adjacent surfaces in the Rhinoceros program. In this way, the first "Surface" block is connected to the lower surface, and the second "Surface" block is connected to the adjacent surface, i.e., the first surface above the lower surface. After that, the "DivideDomain^2" block

is defined. The block consists of three inputs: "I" which indicates the domain; "U" which represents the number of segments in the U direction, and "V" which represents the number of segments in the V direction. The inputs "U" and "V" define how many parts the surface will be divided into in both directions. Two "Sliders," each with a value of 5, are defined since the surface is divided into five equal parts in the x direction (i.e., the U direction) and in the y direction (i.e., the V direction), as shown in Figure 9. The Slider values for "U" and "V" can differ.

Additionally, the space can be divided into many smaller parts. The "Sliders" (with a defined value of 5) are connected to the inputs "U" and "V" of the "DivideDomain²" block. The same procedure is repeated for all subsequent adjacent surfaces using the same principle.

Next, the "BlendBox" is defined, which consists of four inputs: "Sa" which indicates Surface A, "Da" which indicates Domain A, "Sb" which indicates Surface B, and "Db" which indicates Domain B. Label A in this case refers to the surface that is at a higher height (upper surface), while the label B refers to the adjacent surface below (lower surface). Therefore, the "Surface" of the upper surface connects to the input "Sa", and the "Surface" of the lower surface connects to the input "Sb". The output of the previously defined "DivideDomain²" block, called "S" for Segments, is connected to the input "D." The output of the "DivideDomain²" "S" of the upper surface (that is, surface A) is connected to the input of the "BlendBox" called "Da", and the output of the "DivideDomain²" "S" of the lower surface (that is, surface B) is connected to the input of the "BlendBox" called "Db", as shown in Figure 9.

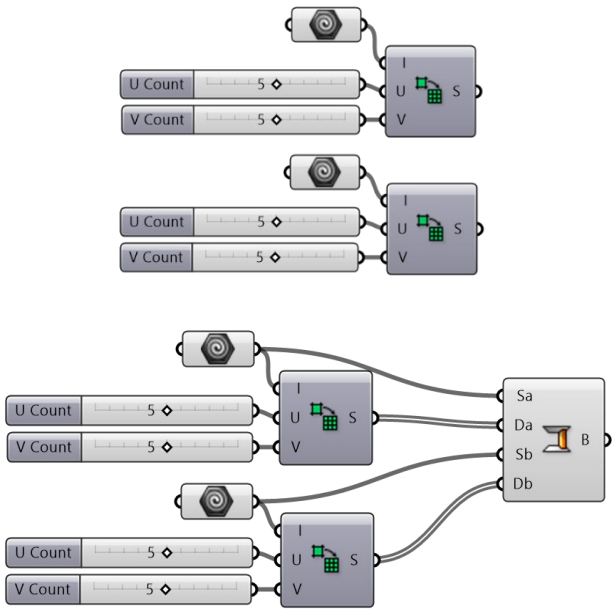

Figure 9 The process of defining a "DivideDomain²" (up) and defining a "BlendBox" (down)

In the Rhinoceros program, the volume between two adjacent surfaces is divided into five and seven equal parts in the U direction and five and seven equal parts in the V direction, resulting in 25 small cubes for a rectangular spatial distribution and 35 small cubes for a parabolic spatial distribution, which are embedded in the volume between the two adjacent surfaces. The procedure is repeated as many times as there are adjacent surfaces in the volume. In the Grasshopper program, the wire model from the Rhinoceros program is first assigned a circular cross-section of nodes via the "Multipipe" command for both lattice structures with a radius of 1.43 mm by connecting the "Slider" value 1.43 to the "NodeSize" input. In Grasshopper, the "Box" block is also defined via the origin coordinate 0,0, then the second coordinate 10,10 (defined square area), and by assigning a height value of 10 mm. The preview in Rhinoceros shows how much the unit cell extends outside the 10mm x 10mm x 10mm dimensions. To successfully fit the unit cell into the 10 mm x 10 mm x 10 mm dimensions, the "SolidIntersection" block is used to cut off the parts of the simple cubic unit cell that go outside the "Box" (Figure 10).

Using the "Box Morph" command, a Simple Cubic unit cell can be embedded within the divided volume. Since the relative density is calculated as the proportion of the volume of the lattice structure to

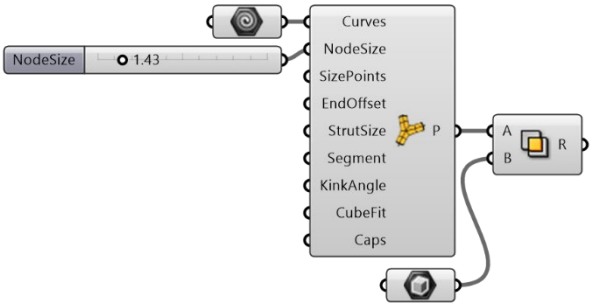

Figure 10 "SolidInterection" block

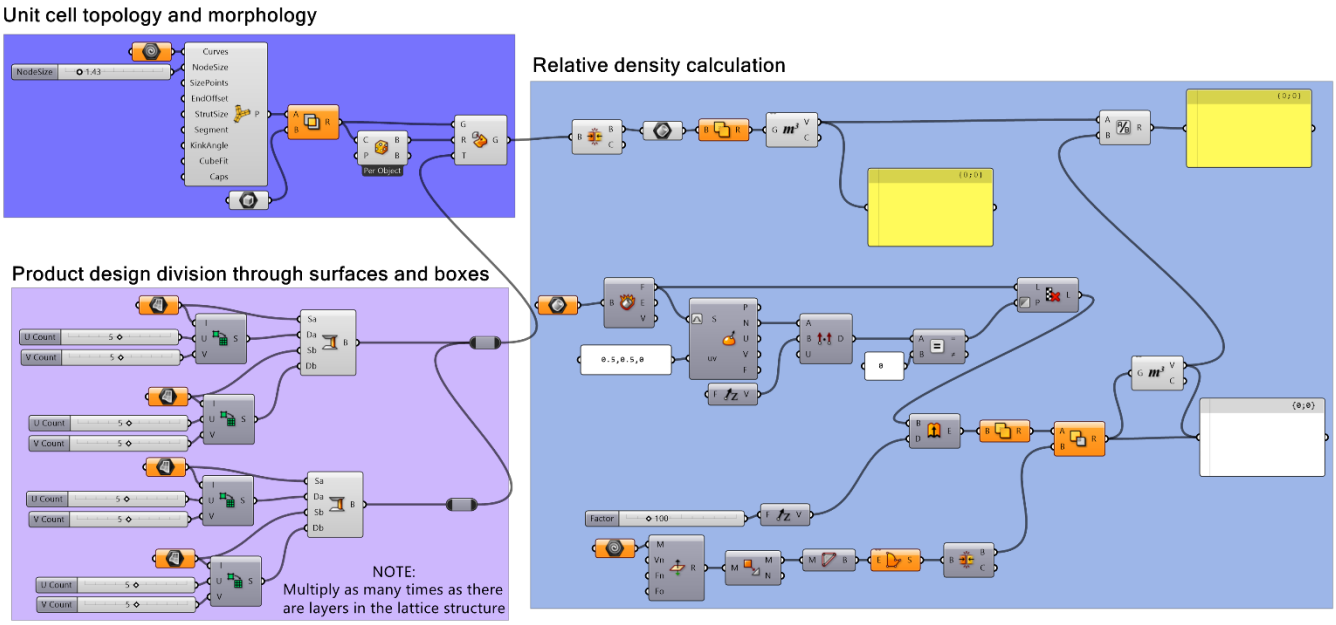

Figure 11 A complete Grasshopper algorithm for generating lattice structures with different morphological distribution of unit cells and constant relative density

the total volume, the Grasshopper algorithm was designed accordingly. By using the generated algorithm (Figure 11) for determining the relative density of both lattice structures (value of 0.201732), it was proven that the defined node radius achieved the target value of the relative density (0.2). The node radius should be reduced if the relative density value exceeds the predicted value. Conversely, if the relative density value falls below the expected value, the node radius should be increased. The relative density of 0.2 was chosen due to the assumed potential for high-quality printing of lattice structures, as a lower relative density also signifies a smaller unit cell node radius. Thin struts are very difficult to print accurately using the additive process of material extrusion layer by layer without distorting the appearance of the unit cells. A higher relative density (e.g., 0.4, 0.5, etc.) means lower porosity; therefore, the dimensions of the cross-section of the lattice structure would be more significant, and the unit cell would lose recognizable shape. Consequently, the lattice structure would then lose the characteristic of a porous structure. A lower relative density would also result in smaller dimensions of the cross-sections, which would most likely distort the final appearance of the lattice structure during printing. The final model of the cubic sample (after "Boolean union") with embedded Simple Cubic unit cells, according to the rectangular spatial distribution, is shown in Figure 12 (left) and according to the parabolic morphological distribution in Figure 12 (right). This demonstrates how the designed algorithm can define different morphological distributions of unit cells within the lattice structure while maintaining the same relative density. It has been proven that, with the Grasshopper extension, it is possible to define an algorithm that enables fitting unit cells in lattice structures on curved surfaces. It is crucial to consider additive manufacturing when designing products for additive technology during the earlier stages of product modeling, as done in the research [21].

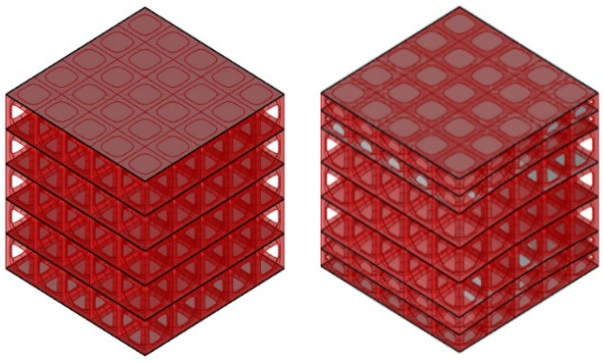

Figure 12 Filling the distances between the flat (left) and parabolic (right) surfaces with Simple Cubic unit cells

5 Conclusion

This paper explains step by step the different morphological distributions of unit cells in lattice structure using the Rhinoceros program and adding the Grasshopper extension but without external plugins. Since the previous research did not explain in detail the various morphological distributions of unit cells in lattice structures, the problem of identifying the crucial steps needs to be carefully studied for successfully incorporating the different morphological distributions of unit cells in lattice structures' final product design. The parameters affecting the relative density of the unit cell are explained in detail, and the unwanted bubble effect is avoided. With this Grasshopper algorithm, it is possible to successfully integrate different morphological distributions of various topologies of unit cells into curved product designs to achieve comfortable lattice structures, which is the goal of future research.

Conflicts of Interest: The author reports there are no competing interests to declare;

References

[1] J. Hulme, A. H. Sakhaei, and M. Shafiee, "Mechanical analysis and additive manufacturing of 3D-printed lattice materials for bone scaffolds," *Mater. Today Proc.*, vol. 101, p. 30-37, 2024, doi: 10.1016/j.matpr.2023.02.278.

[2] N. Namvar, I. Moloukzadeh, A. Zolfagharian, F. Demoly, and M. Bodaghi, "Bio-inspired design, modeling, and 3D printing of lattice-based scale model scooter decks," *Int. J. Adv. Manuf. Technol.*, vol. 126, no. 7–8, p. 2887–2903, 2023, doi: 10.1007/s00170 023-11185-8.

[3] N. Khan and A. Riccio, "A systematic review of design for additive manufacturing of aerospace lattice structures: Current trends and future directions," *Prog. Aerosp. Sci*., vol. 149, no. June, p. 101021, 2024, doi: 10.1016/j.paerosci.2024.101021

[4] A. Nazir, K. M. Abate, A. Kumar, and J. Y. Jeng, "A state-of-the-art review on types, design, optimization, and additive manufacturing of cellular structures," *Int. J. Adv. Manuf. Technol.*, vol. 104, no. 9–12, p. 3489–3510, 2019, doi: 10.1007/s00170-019-04085-3.

[5] M. Al Rifaie, H. Abdulhadi, and A. Mian, "Advances in mechanical metamaterials for vibration isolation: A review," *Adv. Mech. Eng.*, vol. 14, no. 3, p. 1–20, 2022, doi: 10.1177/16878132221082872.

[6] S. Ghaemi Khiavi, B. Mohammad Sadeghi, and M. Divandari, "Effect of topology on strength and energy absorption of PA12 non-auxetic strut-based lattice structures," *J. Mater. Res. Technol.*, vol. 21, p. 1595–1613, 2022, doi: 10.1016/j.jmrt.2022.09.116.

[7] I. Maskery et al., "An investigation into reinforced and functionally graded lattice structures," *J. Cell. Plast.*, vol. 53, no. 2, p. 151–165, 2017, doi: 10.1177/0021955X16639035.

[8] D. S. J. Al-Saedi and S. H. Masood, "Mechanical Performance of Functionally Graded Lattice Structures Made with Selective Laser Melting 3D Printing," *IOP Conf. Ser. Mater. Sci. Eng.*, vol. 433, no. 1, 2018, doi: 10.1088/1757-899X/433/1/012078.

[9] L. Bai et al., "Mechanical properties and energy absorption capabilities of functionally graded lattice structures: Experiments and simulations," *Int. J. Mech. Sci.*, vol. 182, no. February, p. 105735, 2020, doi: 10.1016/j.ijmecsci.2020.105735.

[10] F. Liu, M. Chen, L. Wang, Z. Xiang, and S. Huang, "Lightweight and Customized Design via Conformal Parametric Lattice Driven by Stress Fields," *Springer Nature Singapore,* vol. Part F2072., 2024. doi: 10.1007/978-981-99-8405-3_12.

[11] P. Dal Fabbro et al., "Analysis of a preliminary design approach for conformal lattice structures," *Appl. Sci.*, vol. 11, no. 23, 2021, doi: 10.3390/app112311449.

[12] A. García-Domínguez, J. Claver, and M. A. Sebastián, "Infill optimization for pieces obtained by 3D printing," Procedia *Manuf.*, vol. 41, p. 193–199, 2019, doi: 10.1016/j.promfg.2019.07.046.

[13] S. Geyer and C. Hölzl, "Comparison of CAD Software for Designing Cellular Structures for Additive Manufacturing," *Appl. Sci.*, vol. 14, no. 8:3306, 2024, doi: 10.3390/app14083306.

[14] M. Mahmoudi, S. A. M. Ghannadpour, and K. H. Nedjad, "Functionally graded multi-morphology lattice structures as an optimized sandwich core via digital light processing additive manufacturing," *Mater. Des.*, vol. 238, no. January, p. 112710, 2024, doi: 10.1016/j.matdes.2024.112710.

[15] F. Liu *et al.*, "Stress-driven generative design and numerical assessment of customized additive manufactured lattice structures," *Mater. Des.*, vol. 241, no. March, p. 112956, 2024, doi: 10.1016/j.matdes.2024.112956.

[16] M. He, L. Yang, C. Zhao, R. Zhang, G. Han, and L. Hao, "Effects of design and manufacturing deviations on compressive properties of glass sponge lattice structures manufactured by selective laser melting," *Mater. Des.*, vol. 242, no. December 2023, p. 113009, 2024, doi: 10.1016/j.matdes.2024.113009.

[17] K. Ferreira, N. Anwer, and C. Mehdi-Souzani, "Characterization of L-PBF lattice structures geometric defects," *Procedia CIRP*, vol. 100, no. March, p. 846–851, 2021, doi: 10.1016/j.procir.2021.05.033.

[18] P. Parenti *et al.*, "Pure copper extrusion Additive Manufacturing of lattice structures for enabling enhanced thermal efficiency in hydrogen production," *Manuf. Lett.*, vol. 41, no. 2023, pp. 1080-1091, 2023, doi: 10.1016/j.mfglet.2024.09.133.

[19] I. Krešić, J. Kaljun, and N. Rašović, "Controlling the Mechanical Response of Stochastic Lattice Structures Utilizing a Design Model Based on Predefined Topologic and Geometric Routines," *Appl. Sci.*, vol. 14, no. 14, 2024, doi: 10.3390/app14146048.

[20] K. M. Park and Y. S. Roh, "Design Optimization of Additive Manufactured Edgeless Simple Cubic Lattice Structures under Compression," *Materials (Basel).*, vol. 16, no. 7:2870, 2023, doi: 10.3390/ma16072870.

[21] N. Rašović, "Recommended layer thickness to the powder-based additive manufacturing using multi-attribute decision support," *Int. J. Comput. Integr. Manuf.*, vol. 34, no. 5, pp. 455–469, 2021, doi: 10.1080/0951192X.2021.1891574.