INTERNATIONAL CONFERENCE ON ENGINEERING DESIGN, ICED23
24-28 JULY 2023, BORDEAUX, FRANCE

GEOMETRIC MODELLING OF HETEROGENEOUS LATTICE STRUCTURES THROUGH FUNCTION REPRESENTATION WITH LATTICEQUERY

Letov, Nikita;
Zhao, Yaoyao Fiona

McGill University

ABSTRACT

Lattice structures are lightweight and possess other unique mechanical and physical properties. Additive manufacturing techniques are often used to produce these structures. Additive manufacturing provides manufacturing freedom that significantly surpasses the one provided by conventional subtractive manufacturing. However, a gap exists between the manufacturing freedom and the geometric modelling freedom in additive manufacturing: it can be extremely challenging to model the designed part because of its high geometric complexity, such as heterogeneous lattice structures. While several tools on the market allow geometric modelling of such structures available on the market, the customization of lattice parameters can still be significantly improved. Moreover, no open-source tools exist to address this issue or to model lattice structures in general. This work presents a novel open-source library for the geometric modelling of lattice structures with customized parameters. The parameter customization is enabled with the function representation approach.

Keywords: Additive Manufacturing, Functional modelling, Computer Aided Design (CAD), Design methods

Contact:
Letov, Nikita
McGill University
Canada
nikita.letov@mail.mcgill.ca

Cite this article: Letov, N., Zhao, Y. F. (2023) 'Geometric Modelling of Heterogeneous Lattice Structures through Function Representation with Latticequery', in *Proceedings of the International Conference on Engineering Design (ICED23),* Bordeaux, France, 24-28 July 2023. DOI:10.1017/pds.2023.205

1 INTRODUCTION

Lattice structures are complex geometric structures of periodic or quasi-periodic unit cells tessellated according to a specific pattern. Before the introduction of additive manufacturing (AM), manufacturing these objects was nearly impossible with conventional subtractive techniques (Li et al., 2006). However, engineering geometric modelling tools such as computer-aided design (CAD) packages have only recently begun advancements to support lattice structures (Savio et al., 2018). Lattice structures can be classified as homogeneous and heterogeneous. Homogeneous lattice structures have their geometric parameters constant across the whole structure. The geometric modelling of such structures is straightforward with most modern geometric modelling approaches. Heterogeneous lattice structures have varying geometric parameters, allowing them to possess different mechanical and physical properties in other structure regions (Liu et al., 2021). However, the geometric modelling of such structures is still limited because the CAD industry has not yet fully adjusted to the transition from subtractive manufacturing to AM (Letov et al., 2021).

Several solutions to the limited design freedom problem have been introduced. Many of these solutions have been reviewed in the literature (Letov et al., 2021; Liu et al., 2021). However, most solutions do not allow the variation of many other geometric parameters of the lattice. For example, some beam-based topologies, such as the truncated cube topology, may require the truncation size as a parameter among the unit cell size and the beam thickness. While the thickness of lattice structures is a geometric parameter commonly considered for variation in heterogeneous lattice structures, the shape of the lattice cross-section of a lattice beam is not widely modelled. Variation of such a parameter affects other geometric properties, such as the cross-section area A and its moment of inertia I, which affect the mechanical properties of lattice structures (Muir et al., 1995).

It has been shown that open-source geometric modelling tools can enhance the design for AM experience. At the same time, the literature review identified an overall lack of free and open-source (FOSS) applications for engineering in general and design in AM in particular (Junk and Kuen, 2016). This lack of FOSS applications results in a lack of lean tools that are developed in an agile manner and accessible to engineering designers (Brasseur, 2018).

This work presents the packaging of the method for the geometric modelling of heterogeneous lattice structures developed in the previous work (Letov and Zhao, 2022) in a free and open-source library called LatticeQuery. The methodology is simplified to be used as a collection of simplified functions, which can generate a heterogeneous lattice structure based on the function arguments. The utilisation of these functions is inspired by the function representation (F-rep) geometric modelling approach. F-rep allows the geometric modelling of highly complex geometric objects, given that a mathematical function that describes their geometry is known (Pasko et al., 1995). Furthermore, it was shown that F-rep is applicable to the geometric modelling of lattice structures, as their geometry can be defined simply by periodic functions (Pasko et al., 2011).

The library presented in this work aims to provide flexible F-rep methods that reduce the amount of user input to provide a desired heterogeneous lattice structure. Figure 1 shows examples of heterogeneous lattice structures that can be modelled with LatticeQuery, and Figure 2 demonstrates the main window of the software prototype, which utilises LatticeQuery.

This work is organised as follows. The related work is reviewed in Section 2. Section 3 presents the proposed geometric modelling approach to be incorporated into the library. The library itself and its technical characteristics are provided in Section 4. The conclusions are presented in Section 5.

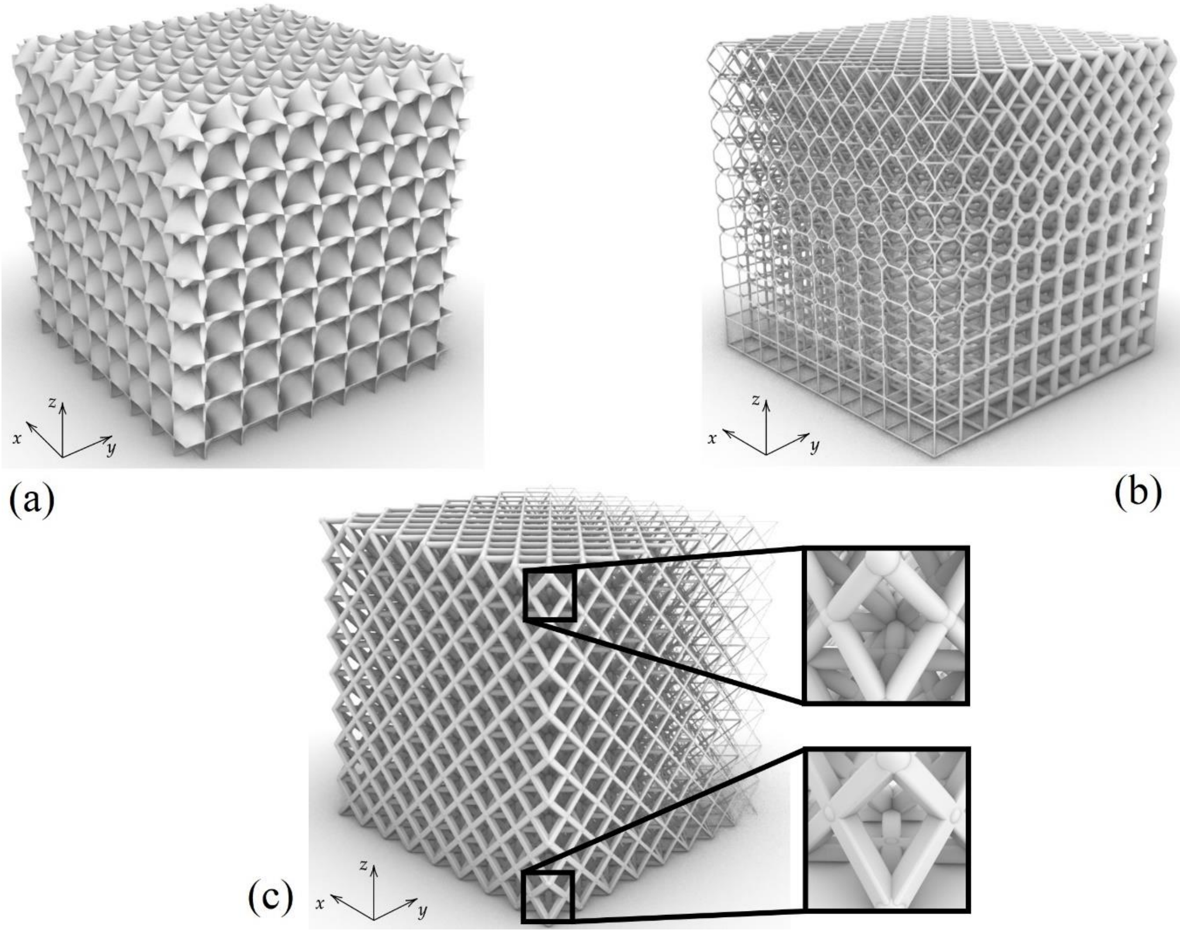

Figure 1. Solid models of lattice structures modelled with LatticeQuery: (a) a Schwarz Primitive lattice structure with a thickness that varies along the z-axis; (b) a truncated cube lattice structure with a thickness that varies along the y-axis and the truncation that varies along the z-axis; and (c) a face-centred cubic lattice structure with the thickness that varies along the y-axis and the beam cross-section changing from square to circle along the z-axis

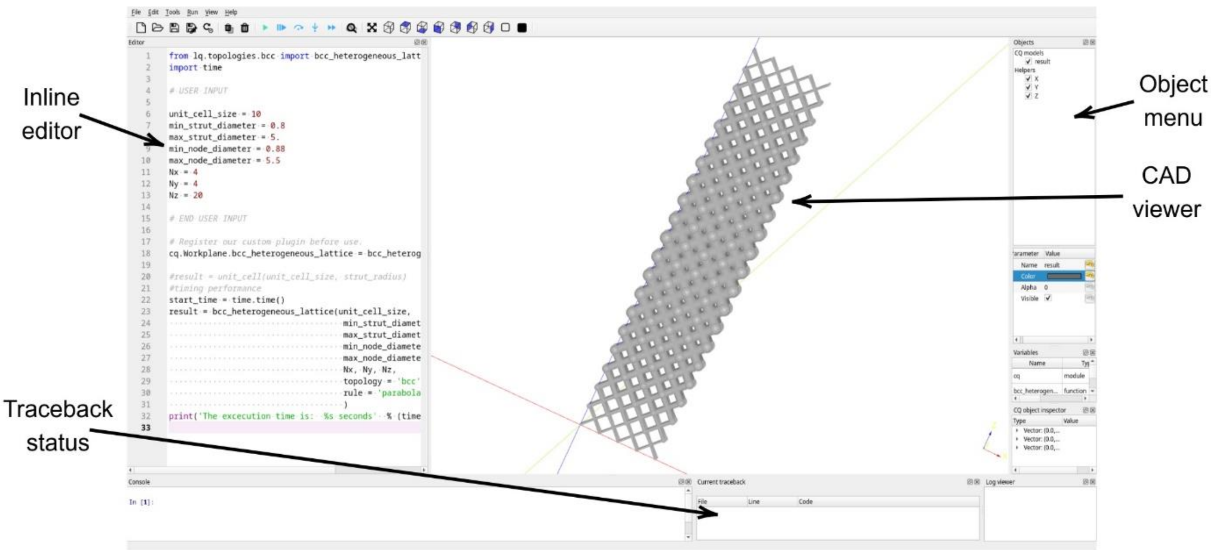

Figure 2. The main window of the software prototype that embeds LatticeQuery

https://doi.org/10.1017/pds.2023.205 Published online by Cambridge University Press

2 RELATED WORK

Geometric solid modelling is integral to creating and documenting an engineering design such as periodic lattice structures. Geometric modelling is crucial in a product lifecycle's conceptual and design stages. These two stages account for approximately 85% of the committed costs of a product, while only 15% are spent on them (Hamelin et al., 2010). While AM significantly enhances manufacturing freedom and allows the manufacturing of lattice structures, design freedom should also be improved to support AM.

Several commercial solutions and prototypes exist to support the geometric modelling of heterogeneous lattice structures, such as Autodesk Netfabb (Autodesk Inc., 2017), Sulis Lattice (Gen3D Ltd., 2019), nTopology (nTopology Inc., 2017), and Siemens NX (Siemens AG, 2007), as well as several plugins for Rhinoceros 3D (Robert McNeel and Associates, 2020) such as Intralattice (Kurtz, 2015), Crystallon (F EQUALS F LLC, 2019), and Dendro (ECR Labs, 2018). However, it was found that many of these solutions provide a linear variation of thicknesses in a specific direction and do not support more complex functionally graded variations. Some solutions, such as nTopology, support non-linear variation of the lattice thickness through implicit modelling. Nevertheless, parametrisation of geometric properties other than lattice thickness remains unexplored. A more detailed review of related work has been performed in preceding works (Letov et al., 2021; Liu et al., 2021). The method proposed in the preceding work is capable of geometric modelling of lattice structures with other non-linearly varying parameters, such as the truncation in Figure 1b and the beam cross-section shape in Figure 1c (Letov and Zhao, 2022, 2023).

However, the approach lacked a graphical user interface negatively impacting its usability. Heterogeneous lattice structures require a substantial number of parameters to be defined. Tools such as Autodesk Netfabb provide a GUI similar to a conventional feature-based CAD package, which increases interactivity but negatively impacts the design intent when designing complex geometry (Mathur et al., 2020). On the other hand, the previously mentioned plugins for Rhinoceros 3D support the geometric modelling of lattice structures through Grasshopper 3D (Davidson, 2009), which introduces a visual programming language (VPL) to describe the geometry as a dataflow diagram. The GUI of nTopology allows the user to design lattice structures with subsequential functional blocks, which is another VPL form. While it was found that a GUI based on a VPL is beneficial for beginners, more geometrically complex structures can become extremely challenging to be designed, thus disrupting the intent in the case of an advanced design (Saito et al., 2017).

While optimising the material spent during an AM process is an important research topic in geometric modelling and topology optimisation for AM (Gopsill et al., 2018; Lu et al., 2014), this work focuses solely on the geometry representation of lattice structures. Nevertheless, geometry representation software can potentially enhance these research directions.

3 A GEOMETRIC MODELLING APPROACH

As mentioned in Section 1, the method for the geometric modelling of heterogeneous lattice structures has been developed in the previous work (Letov and Zhao, 2022). The developed software prototype included the implementation of the methodology with an open-source geometric modelling kernel (GMK). However, the usability of the software prototype was heavily limited due to a lack of a usable GUI. This work describes the implementation of the previously developed methodology into a Python library and presents a proper usable GUI to work with that library.

3.1 Methodology

The method takes its inspiration from the F-rep technique, which defines a solid body $\mathbb{S}$ with a real-valued function F as

$$F(\mathbf{X}) \geq 0, \tag{1}$$

where $\mathbf{X} = (x, y, z) \subset \mathbb{R}^3$ is the entire design space, and F is defined such that $F(\mathbf{X}) = 0$ is the set of points forming the boundary $\partial\mathbb{S}$ of the solid body, which is an oriented 2-manifold and can be defined as

https://doi.org/10.1017/pds.2023.205 Published online by Cambridge University Press

$$\partial\mathbb{S} \coloneqq \mathbb{S} \cap (\mathbf{X} - \mathbb{S}) = \{\mathbf{X}|F(\mathbf{X}) = 0\} \tag{2}$$

and $F(\mathbf{X}) \geq 0$ is a set of points belonging to the solid body, or

$$\mathbb{S} \coloneqq \cap \{\mathbf{X}|F(\mathbf{X}) \geq 0\}. \tag{3}$$

The method, however, expands the definition of a solid body in Equation 1 to provide an enhanced description of a lattice structure. This definition is achieved by separating the function F into two functions that describe the topology T of the lattice and the geometric parameters P of the lattice. The resulting solid model of a lattice structure is defined in this work as a composition of these two functions, or

$$F(\mathbf{X}) = (P \circ T)(\mathbf{X}) \geq 0. \tag{3}$$

This approach allows the definition of the parameters and the topology of a lattice separately, thus simplifying the definition of the solid model of the lattice. For example, consider a lattice structure with the body-centred cubic (BCC) topology with $4 \times 4 \times 20 = 320$ cubic unit cells. The topology T in this approach is defined as a skeletal graph. Such an approach reduces the complexity associated with defining the topology by reducing the number of geometric parameters needed to obtain the general shape of the topology (Lee et al., 2012). For the beam-based topologies, T can be defined as a set of lines corresponding to the topology beams intersecting at nodes. In this example, T for the BCC topology can be defined as a set of lines for the diagonals of a cube, or

$$T(\mathbf{X}):\begin{bmatrix} x = y = z, \\ -x + u = y = z, \\ x = y = -z + u, \\ -x + u = y = -z + u, \end{bmatrix} \tag{3}$$

where u is the unit cell size. The skeletal graph for the BCC topology is shown in Figure 3.

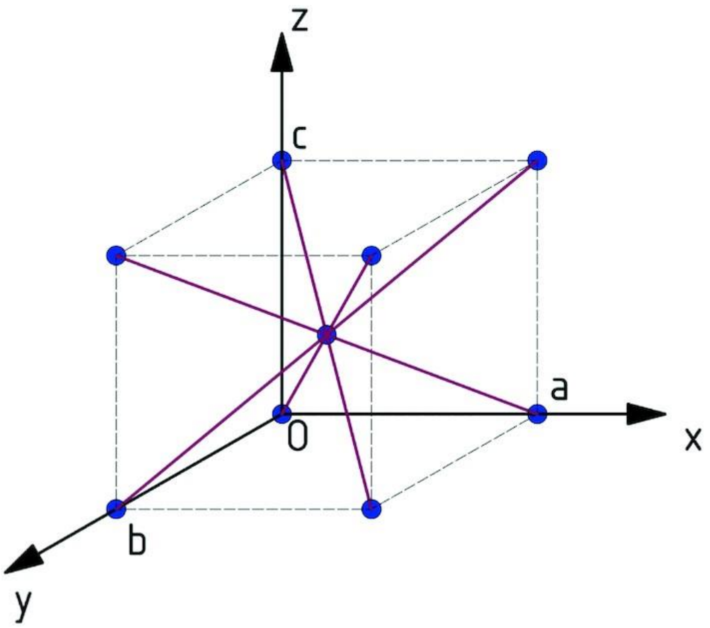

Figure 3. The skeletal graph of the BCC topology. In the general case, the unit cell is cuboid with the sides a, b and c

Let the unit cell size be $u = 10$ mm, and the thickness t of the lattice be defined by the following parabolic distribution:

$$P(\mathbf{X}):\ t(z) = -16.8\mathrm{z}^2 + 16.8\mathrm{z} + 0.8, \tag{4}$$

which is sketched in Figure 4. Note that z is defined here for the interval $z \in [0,1]$ and is then mapped to the integer range $z_u \in [1, N_z] \subset \mathbb{N}^+$, which corresponds to the unit cell distribution. The resulting lattice structure, in this case, takes the form shown in Figure 2.

https://doi.org/10.1017/pds.2023.205 Published online by Cambridge University Press

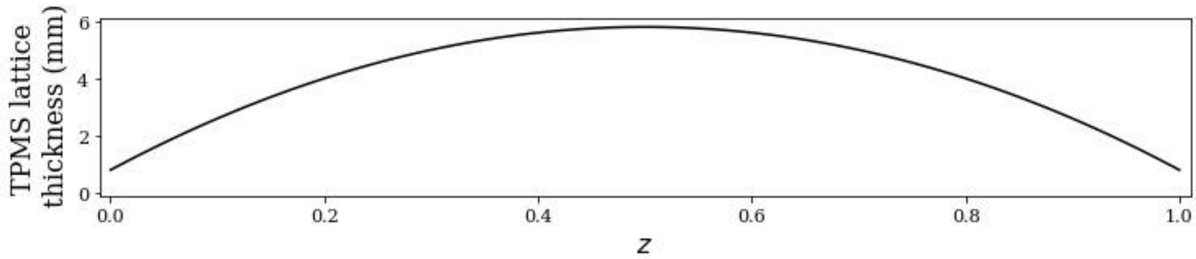

Figure 4. The parabolic distribution of thickness

Note that the topologies based on triply periodic minimal surfaces (TPMS) can also be described with this approach since the topology T can be defined as the equation that defines the surface. For example, Figure 1a shows a heterogeneous lattice structure based on the Schwarz Diamond (Schwarz D) surface, which is described by its approximate implicit surface equation

$$T(\mathbf{X}): \sin(x)\sin(y)\sin(z) + \sin(x)\cos(y)\cos(z) + \cos(x)\sin(y)\cos(z) + \cos(x)\cos(y)\sin(z) = 0. \quad (5)$$

A notable advantage of this approach is its ability to define the variation of geometric parameters other than thickness. For example, Figure 1b shows a lattice structure with a truncated cube topology. The truncation can be a variable parameter τ that allows a gradual transition from the simple cubic topology to the cuboctahedron topology (Letov and Zhao, 2023). Figure 1c shows a lattice structure with the thickness varying in the y-direction and the beam's cross-section gradually changing from circle to square along the z-axis.
The approach allows geometric modelling of conformal lattice structures by representing the geometry in different coordinate systems, such as the cylindrical. However, further work is required to enable the approach to infill a predefined volume similar to existing solutions that allow geometric modelling of conformal lattices.

The preceding work explains the methodology in more detail (Letov and Zhao, 2022).

3.2 Technical aspects related to the implementation of the methodology

A geometric modelling kernel (GMK) is a software package located at the core of any existing CAD and is responsible for providing a program interface to the mathematical theorems and rules that define 3D geometry (Golovanov, 2014). The methodology in this work was implemented using Open CASCADE Technology (OCCT) (Open Cascade, 2018) – the most widely used open-source GMK with extensive documentation.
The software architecture's simplicity and the rapid and lean development of the software prototype have been identified as the core characteristics of the software prototype development. Thus, it was decided to use CadQuery (Urbańczyk, Wright, Cowden, et al., 2021) – a Python library that allows parametric modelling with OCCT in the Python programming language. Working with lattice structures requires dealing with large arrays of data. Therefore, NumPy (Harris et al., 2020) was used to operate the arrays of geometric parameters and interpolate geometric curves to support the TPMS topologies.

The supported topologies include beam-based topologies such as simple cubic, BCC, and face-centred cubic (FCC), as well as several variations of these topologies such as self-supporting FCC without horizontal beams (S-FCC), BCC with additional four z-direction oriented beams (BCCz), FCC with additional four z-direction oriented beams (FCCz), S-FCCz, face- and body-centred cubic (FBCC), S-FBCC, and S-FBCCz. Other supported beam-based topologies include diamond, rhombicuboctahedron, and truncated cube topologies. TPMS-based topologies are presented by gyroid, Schwarz Primitive (Schwarz P), and Schwarz D surfaces.

4 THE DEVELOPED LIBRARY

The LatticeQuery library was developed to be used from within CQ-editor (Urbańczyk, Wright, Ebner, et al., 2021) – an inline editor with a CAD viewer developed for CadQuery. The organisation of the

https://doi.org/10.1017/pds.2023.205 Published online by Cambridge University Press

methodology is presented in this section. The software is developed to make it modular and easily extendable, as well as simplicity of usage.

4.1 Architecture

The inline editor is used to provide commands to the LatticeQuery library. LatticeQuery uses parametric methods from CadQuery to access the low-level functionality of OCCT and to generate a lattice structure according to the user commands. CadQuery can send status messages from OCCT to the traceback status menu. The resulting geometry is then presented in the CAD viewer and can be exported in a CAD file via the object menu. OCCT supports exports in stereolithography (STL) and STEP file formats. The software architecture is illustrated in Figure 5.

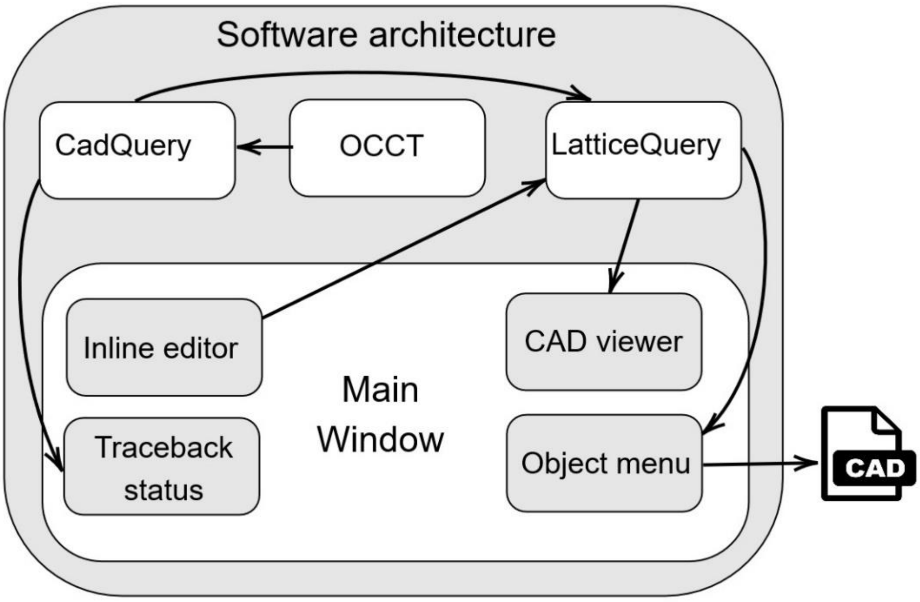

Figure 5. The high-level software architecture of LatticeQuery

The library uses the OCCT instructions provided by CadQuery to generate the geometry of the predefined topologies, which can be called from the inline editor. Several standard functions are shared between the topologies. These functions include the generation of the design space subdivided for all unit cells, the generation of beams by two points, and the cross-section parameters. The library architecture is presented in Figure 6.

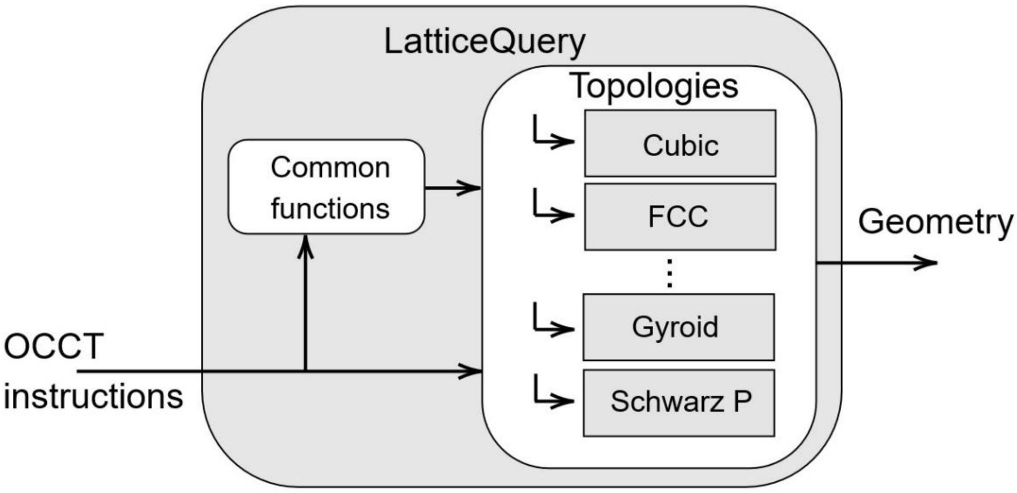

Figure 6. The low-level software architecture of the LatticeQuery library

The resulting library is published under a permissive license and is accessible as a FOSS (Letov, 2022).

https://doi.org/10.1017/pds.2023.205 Published online by Cambridge University Press

4.2 Usage

As an example of the application of the algorithm, consider the BCC heterogeneous lattice structure that is shown in Figure 2, which is modelled with LatticeQuery. The input parameters remain the same as in the example presented in Section 2, and the solid model can be obtained with the programming script presented in Figure 7.

```
from lq.topologies.bcc import bcc_heterogeneous_lattice
# USER INPUT
unit_cell_size = 10
min_strut_diameter = 0.80
max_strut_diameter = 5.0
min_node_diameter = 0.88
max_node_diameter = 5.50
Nx = 4
Ny = 4
Nz = 20
# END USER INPUT
# Register the custom plugin before use.
cq.Workplane.bcc_heterogeneous_lattice = bcc_heterogeneous_lattice
result = bcc_heterogeneous_lattice(unit_cell_size,
                                   min_strut_diameter,
                                   max_strut_diameter,
                                   min_node_diameter,
                                   max_node_diameter,
                                   Nx, Ny, Nz,
                                   topology = 'bcc',
                                   rule = 'parabola'
                                   )
```

Figure 7. An example of a program that generates a lattice structure with LatticeQuery

In this case, the minimum and maximum values of the beam diameter are specified. Note that the lattice nodes are represented as spheres with their diameter following a similar parabolic distribution. The topology is set to BCC, with other possible variations of the BCC topology being BCCz, S-BCC, and S-BCCz.

The lattice structure presented in Figure 2 took 194.61 s to generate on a machine running a Linux OS equipped with an NVIDIA® RTX™ 3060 Laptop graphical processing unit, an AMD Ryzen™ 5900HX central processing unit, and a solid-state disk drive. Further analysis of the computational efficiency is presented in the preceding work (Letov and Zhao, 2022). The resulting model can be exported in STL or STEP file format.

5 CONCLUSIONS

This work presents a free and open-source library that can model heterogeneous lattice structures for AM. The heterogeneity of lattice structures modelled with this tool is expressed by the variation of the lattice thickness, beam cross-section, and truncation. The library is released as a FOSS and is made available to the open-source community. Releasing LatticeQuery as open-source may encourage the open-source community to contribute to developing the presented software and FOSS in engineering and AM in general.

For future work, it is proposed to provide an interface that would allow the import of any user-defined function. Such functionality would enable higher design freedom. It is also proposed to perform a study of applying this tool in several design processes.
The current version of the software has a single-document interface. At the same time, many modern CAD software packages possess a multi-document interface (MDI) that allows operating with multiple tabs of documents simultaneously. It is proposed to switch to the MDI for future versions of LatticeQuery.

https://doi.org/10.1017/pds.2023.205 Published online by Cambridge University Press

ACKNOWLEDGEMENTS

This research work is supported by the National Sciences and Engineering Research Council of Canada Discovery Grant RGPIN-2018-05971 and the McGill Engineering Doctoral Award.

REFERENCES

Autodesk Inc. (2017), "Fusion 360 with Netfabb", available at: https://www.autodesk.com/products/netfabb/overview (accessed 7 February 2023).

Brasseur, V.M. (2018), *Forge Your Future with Open Source: Build Your Skills. Build Your Network. Build the Future of Technology*, Pragmatic Bookshelf, Raleigh, North Carolina.

Davidson, S. (2009), "Grasshopper – Algorithmic modeling for Rhino", available at: https://www.grasshopper3d.com (accessed 13 February 2023).

ECR Labs. (2018), "Dendro", available at: https://www.ecrlabs.com/dendro (accessed 7 February 2023).

F EQUALS F LLC. (2019), "f=f: Crystallon", available at: http://fequalsf.blogspot.com/p/crystallon.html (accessed 7 February 2023).

Gen3D Ltd. (2019), "Sulis AM software", available at: https://gen3d.com/software (accessed 7 February 2023).

Golovanov, N. (2014), *Geometric Modeling*, CreateSpace Independent Publishing Platform, available at: https://books.google.ca/books?id=4jUgrgEACAAJ.

Gopsill, J.A., Shindler, J. and Hicks, B.J. (2018), "Using finite element analysis to influence the infill design of fused deposition modelled parts", *Progress in Additive Manufacturing*, Vol. 3 No. 3, pp. 145–163. https://doi.org/10.1007/s40964-017-0034-y.

Hamelin, R.D., Walden, D.D. and Krueger, M.E. (2010), "4.4.2 INCOSE Systems Engineering Handbook v3.2: Improving the Process for SE Practitioners", *INCOSE International Symposium*, Vol. 20 No. 1, pp. 532–541. https://doi.org/10.1002/j.2334-5837.2010.tb01087.x.

Harris, C.R., Millman, K.J., van der Walt, S.J., Gommers, R., Virtanen, P., Cournapeau, D., Wieser, E., et al. (2020), "Array programming with NumPy", *Nature*, Vol. 585 No. 7825, pp. 357–362. https://doi.org/10.1038/s41586-020-2649-2.

Junk, S. and Kuen, C. (2016), "Review of open source and freeware CAD systems for use with 3D-printing", *Procedia CIRP*, Vol. 50, pp. 430–435. https://doi.org/10.1016/j.procir.2016.04.174.

Kurtz, A. (2015), "Intralatice", available at: http://www.intralattice.com (accessed 7 February 2023).

Lee, J.-H., Wang, L., Boyce, M.C. and Thomas, E.L. (2012), "Periodic bicontinuous composites for high specific energy absorption", *Nano Letters*, Vol. 12 No. 8, pp. 4392–4396. https://doi.org/10.1021/nl302234f.

Letov, N. (2022), "jalovisko/LatticeQuery", GitHub.

Letov, N., Velivela, P.T., Sun, S. and Zhao, Y.F. (2021), "Challenges and opportunities in geometric modeling of complex bio-inspired three-dimensional objects designed for additive manufacturing", *Journal of Mechanical Design*, Vol. 143 No. 12, p. 121705. https://doi.org/10.1115/1.4051720.

Letov, N. and Zhao, Y.F. (2022), "A geometric modelling framework to support the design of heterogeneous lattice structures with non-linearly varying geometry", *Journal of Computational Design and Engineering*, Vol. 9 No. 5, pp. 1565–1584. https://doi.org/10.1093/jcde/qwac076.

Letov, N. and Zhao, Y.F. (2023), "Beam-based lattice topology transition with function representation", *Journal of Mechanical Design*, Vol. 145 No. 1, available at:https://doi.org/10.1115/1.4055950. https://doi.org/10.1115/1.4055950.

Li, L., Syed, W.U.H. and Pinkerton, A.J. (2006), "Rapid additive manufacturing of functionally graded structures using simultaneous wire and powder laser deposition", *Virtual and Physical Prototyping*, Vol. 1 No. 4, pp. 217–225. https://doi.org/10.1080/17452750601141523.

Liu, Y., Zheng, G., Letov, N. and Zhao, Y.F. (2021), "A survey of modeling and optimization methods for multi-scale heterogeneous lattice structures", *Journal of Mechanical Design*, Vol. 143 No. 4, p. 040803. https://doi.org/10.1115/1.4047917.

Lu, L., Sharf, A., Zhao, H., Wei, Y., Fan, Q., Chen, X., Savoye, Y., et al. (2014), "Build-to-last: strength to weight 3D printed objects", *ACM Transactions on Graphics*, Vol. 33 No. 4, pp. 97:1-97:10. https://doi.org/10.1145/2601097.2601168.

Mathur, A., Pirron, M. and Zufferey, D. (2020), "Interactive programming for parametric CAD", *Computer Graphics Forum*, Vol. 39 No. 6, pp. 408–425. https://doi.org/10.1111/cgf.14046.

Muir, P., Johnson, K.A. and Markel, M.D. (1995), "Area moment of inertia for comparison of implant cross-sectional geometry and bending stiffness", *Veterinary and Comparative Orthopaedics and Traumatology*, Vol. 08 No. 03, pp. 146–152. https://doi.org/10.1055/s-0038-1632446.

nTopology Inc. (2017), "Next generation engineering design software", available at: https://ntopology.com (accessed 7 February 2023).

Open Cascade. (2018), "Open CASCADE Technology", available at: https://www.opencascade.com/open-cascade-technology/ (accessed 7 October 2022).

Pasko, A., Adzhiev, V., Sourin, A. and Savchenko, V. (1995), “Function representation in geometric modeling: concepts, implementation and applications”, *The Visual Computer*, Vol. 11 No. 8, pp. 429–446. https://doi.org/10.1007/BF02464333.

Pasko, A., Fryazinov, O., Vilbrandt, T., Fayolle, P.-A. and Adzhiev, V. (2011), “Procedural function-based modelling of volumetric microstructures”, *Graphical Models*, Vol. 73 No. 5, pp. 165–181. https://doi.org/10.1016/j.gmod.2011.03.001.

Robert McNeel and Associates. (2020), “Rhino – Rhinoceros 3D”, available at: https://www.rhino3d.com (accessed 7 February 2023).

Saito, D., Washizaki, H. and Fukazawa, Y. (2017), “Comparison of text-based and visual-based programming input methods for first-time learners”, *Journal of Information Technology Education: Research*, Vol. 16, pp. 209–226. https://doi.org/10.28945/3775.

Savio, G., Meneghello, R. and Concheri, G. (2018), “Geometric modeling of lattice structures for additive manufacturing”, *Rapid Prototyping Journal*, Vol. 24 No. 2, pp. 351–360. https://doi.org/10.1108/RPJ-07-2016-0122.

Siemens AG. (2007), “NX - Siemens Software”, available at: https://www.plm.automation.siemens.com/global/en/products/nx (accessed 14 February 2023).

Urbańczyk, A., Wright, J., Cowden, D., Innovations Technology Solutions, Özderya, H.Y., Boyd, M., Agostini, B., et al. (2021), “CadQuery/cadquery: CadQuery”, Github.

Urbańczyk, A., Wright, J., Ebner, G., Buchanan, J., Boyd, M., Saville, W., Trhoň, A., et al. (2021), “CadQuery/CQ-editor”, Github.