Image-Driven Manufacturing of Graded Lattices by Fused Deposition Modeling

Luca Grigolato[1,2]([✉]) [ID], Stefano Rosso[2] [ID], Enrico Bernardo[1] [ID],
Gianmaria Concheri[2] [ID], and Gianpaolo Savio[2] [ID]

[1] DII Department, University of Padova, Via Marzolo 9, 35131 Padova, Italy
[2] DICEA Department, University of Padova, Via Venezia, 1, 35131 Padova, Italy
luca.grigolato@phd.unipd.it

Abstract. Lattice structures are receiving a renewed interest in many areas such as biomedical and industrial fields, due to the capabilities of additive manufacturing technologies which allow for the fabrication of very complex shapes. Currently, several methods and tools are described in the scientific literature and some commercial software are introducing dedicated packages to reduce designer efforts for lattice structure design and optimization. However, by using commercial CAD/CAM tools in the fabrication of components filled by lattice structure, several critical issues remain and need to be taken into consideration. This work aims at manufacturing variable-density lattice structures via fused deposition modeling deriving the density map from a grayscale or color image. In the proposed approach, the shell-based lattice model is not achieved by CAD tools, but only during the CAM process, while the lattice relative density is computed by editing the G-code, modifying the extrusion flow according to the local grayscale of a volumetric CAD model, defined from an image. The main advantages are related to the absence of a graded lattice geometric model and the consistency of the toolpath. The method is tested on various images and patterns, and can find applications in artworks, embedded information on components, and functional 3D printed parts, such as the replication of the density map of a bone derived from a DICOM grayscale image.

Keywords: Additive manufacturing · Graded lattices · Heterogeneous objects · Fused deposition modeling

1 Introduction

Bridging the gap between geometric model complexity and manufacturing constraints is one of the main challenges to fully exploit the additive manufacturing technologies capabilities. Even if virtual geometric models suitable for powder bed fusion additive manufacturing technologies can be obtained, ensuring a good compromise among design speed, model complexity, and technologies, due to the manufacturing constraints, other technologies such as material extrusion (MEX) shows difficulties in managing the model complexity related to lattice structures and much more when dealing with shell-based lattice structures with variable thickness and/or variable density infill. Computer-aided

design (CAD) software tools are now increasing the capabilities of realizing these types of geometric models, but designer expertise is typically required [1, 2]. In addition, the resulting polygonized surface file can be extremely huge in size, compromising the data exchange between designer and manufacturer in the 3D printing workflow. Indeed, computer-aided manufacturing (CAM) software firstly provide the division of the object in layers, parallel to the printing direction, and then generate the printing toolpath for each layer, computing the machine CNC programming language, i.e., the G-code file. Usually, this process is automatically made by several slicing software, but in complex shapes, the CAM of lattice structures, especially based on thin walls, can generate inconsistent toolpaths, unable to reproduce the CAD model.

This work aims at manufacturing variable-density lattice structures via fused deposition modeling (FDM) deriving the density map from a grayscale image. As mentioned, using commercial CAD/CAM tools this result is difficult or even unobtainable. To overcome this limit, the shell-based lattice structure is not achieved by CAD tools, but only during the CAM process, adopting different infill strategies, while the graded relative density is obtained by editing the G-code, modifying the extrusion flow according to the local grayscale, defined in a volumetric geometric model. The main advantages are related to the absence of the lattice geometric model and the consistency of the toolpath, overcoming the previously highlighted limitations. The two main proposals are to avoid geometric model manipulation and inconsistency in the toolpath. Thus, the objective is to interact between CAD and CAM processes, to take advantage of both.

2   Background

Standard workflow in AM starts from a CAD model and finishes with a 3D printed physical object of the CAD model. In between there are various phases involving CAD modeling, polygonization of the CAD model, slicing operations, and toolpath generation, resulting in a CNC file usually called G-code file [3, 4]. Several approaches for the geometric modeling of lattices, based on different representation schemes, are available in the literature [5–7]. Among these, some approaches present advantages and have been employed more frequently than others. This is the case of direct polygonal mesh modeling and implicit (or function-based) modeling approaches. An example of the first one is given by Savio et al. [8]; as described it is possible to model beam and shell-like structures using low-poly meshes and if needed use a subdivision algorithm to smooth the whole mesh. In this manner minimal surfaces lattices can be also generated by successive approximations and the approach can be extended to graded lattices [9]. On the other hand, implicit modeling represents a flexible solution for 3D modeling, especially for lattice structures [10]. Several studies have shown the potentiality of implicit modeling for generating relative density graded, cell size graded and multi-morphology lattices [11–13]. From an AM point of view, the principal drawback of this approach consists in the conversion, that is needed at some point of the AM workflow, from the function representation into an auxiliary B-rep, such as a mesh (e.g. isosurface polygonization), for subsequent slicing operations. To avoid the auxiliary polygonized representation, the main solutions are the conversion into a stack of raster images, representing each layer in the Z direction, at the machine resolution (which is an acceptable input for some

AM technologies) or directly calculating the machine toolpath from the implicit model [10, 14, 15]. For this reason, some authors investigated the possibility of modeling the machine toolpath directly by using geometric entities different from meshes, such as points and curves, especially in the case of MEX technologies [5].

To create lattice-based functional objects, it is possible to operate on several levels of complexity, and properties related for instance to geometry, material, or process characteristics. The geometric characteristic can be tuned by operating on the unit cell shape, the unit cell size, or the unit cell relative density. The material characteristic can be tuned by modifying the material microstructure or by using more than one material, thus changing the composition over the volume. At last, the process can influence the final object by tuning the manufacturing parameters, for instance, temperature or speed, which influence several characteristics such as actual flow rate, the accuracy of the part, adhesion among layers, and anisotropy.

Above all other geometric characteristics, the relative density can be considered the primary design variable for AM lattices [13]. Since one of the main objectives is the determination of structure-property relationships to control the functionality of a component, the physical properties of lattices are influenced mainly by their relative density, which is reflected in the well-known scaling laws of Gibson and Ashby [16]. Relative density is also equivalent to solid volume fraction, which represents the solid volume of the lattice with respect to the overall volume of the space it occupies.

Several software tools have been developed to allow precise control of the lattice modeling and the AM process for specific applications from bioprinting to ceramics, or to translate vector graphics into toolpath and to post-process G-code [4, 8, 9, 13]. Among all the software tools, a similar idea to the one proposed in this work was described by Moetazedian et al. [17]. The work is based on the concept of avoiding CAD modeling, subsequent slicing, and using continuous deposition paths with varying extrusion width as opposed to fragmented ones with constant extrusion width, to improve the quality of manufactured parts by eradicating voids and defects. To demonstrate the approach in [17], the same research team developed and published a free open-source software called FullControl GCode Designer [4]. In this tool, the user defines every segment of the toolpath along with all printing parameters, which may be related to geometric and non-geometric factors, at all points along the toolpath. Similarly, Silkworm [18] is an open project released as a plugin in Grasshopper/Rhinoceros (McNeel & Associates, USA) capable to translate non-solid geometry, i.e., surfaces, curves, and points and further, into G-code for 3d printing purposes. A tool acting directly on the G-code file by post-processing operations is Velocity Painting [19]. It's a free open-source script written in perl language that, when given an image file and a G-code file, post-processes the G-code to change print speeds according to the intensity of the image pixels, projecting the image onto the skin of the physical model. This tool was used in [17] to generate graded patterns.

Suggested by the weaknesses and the advantages of approaches and software for lattice modeling and manufacturing found in the literature, the proposed method tries to catch the best of both worlds with an 'in between' solution: CAD modeling operations are used for the B-rep macro-shape design, CAM software are used for the infill modeling allowing an easy lattice design, directly on the G-code, and a CAD volumetric approaches

are used for the G-code editing. In this way, high complex graded lattice geometric modeling is bypassed avoiding intricate implicit or B-rep representation.

3   Modeling Method

The proposed method consists of several steps, as Fig. 1 depicts. Firstly, a bulk object is geometrically modeled by a commercial CAD tool. After that, the 3D model is used for two operations: the CNC file (G-code) computation and the discrete volumetric model definition, i.e., a virtual model in which each voxel has a scalar value. On one side, the bulk object is sliced using an open-source CAM software, such as Slic3r, in which, together with process parameters, the infill pattern and density are also defined. Among the infill patterns, gyroid infill or other 3D shell-based shapes are chosen. Infill density, on the other hand, defines the relative density of the pattern and directly influences the unit cell size of the chosen pattern. By controlling these two parameters the shell-based lattice can be computed, in the CAM software directly in the toolpath. As a result, the file containing the CNC programming language for additive manufacturing (G-code file) is generated. On the other side, a volumetric map of scalar values is modeled, which is used to drive the 'thicknesses' of the shell lattice. In this work, the map comes from the colors of a 2D image, which are firstly extracted from the grayscale version of the image and then 'projected' into the volumetric array. This process is accomplished in Grasshopper, a Rhinoceros (McNeel & Associates) plugin. The third step consists in modifying the extrusion parameter 'E' of the infill path in the G-code file. The 'E' values are computed as a function of the volumetric map. A script has been developed in Python language to realize this set of operations. The developed script parses the G-code file, finds the 'E' values of the infill, which represents the amount of extruded material along the toolpath, calculates the new 'E' values based on the image color data, replace the new 'E' values, and finally reassembles the modified G-code file. Finally, the part can be manufactured using a desktop FDM 3D printer with a direct drive configuration to reduce the delay [17]. The calculation of the new 'E' values is based on a trilinear interpolation among the color values stored in the volumetric model, considering the centroid coordinates of the voxels and the actual 'E' values 'stored' at each starting point of the segmented toolpath. To summarize, the following steps are implemented to obtain an image embedded in the part infill:

1. CAD modeling: model the bulk object.
2. CAM computation: generate toolpath and the G-code file.
3. Volumetric modeling: the implementation converts a grayscale image into a voxel-based volumetric model (each voxel has a relative density value in the range 0–1 derived from the projected image).
4. G-code editing: modify the extrude material based on the volumetric model density map.
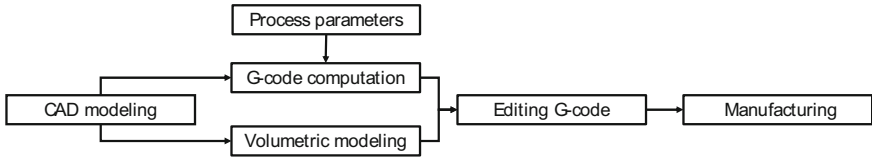5. Manufacturing.

Fig. 1. Flowchart with the macro phases of the proposed method.

3.1 G-code Computation

The G-code is computed by the Slic3r 1.30 software, using the process parameters summarized in Table 1. As infill pattern, the gyroid (sheet-based) was selected due to its quasi-isotropic mechanical behavior [20]. The infill constant density was selected to allow a unitary relative density after the G-code editing process and was established by empirical experiments based on the other process parameters (layer thickness, infill speed, nozzle diameter, extrusion temperature).

Table 1. Constant process parameters

Layer thickness	Infill pattern	Infill speed	Nozzle diameter	Filament diameter	Bed temperature	Extrusion temperature
mm	-	mm/s	mm	mm	°C	°C
0.2	Gyroid	25	0.4	1.75	55	215

3.2 Volumetric Modeling

The volumetric model is constructed starting from a single (2D) image. The RGB image is converted into a vector of normalized grayscale values in the range 0–1 that can be interpreted as a relative density map. This vector is obtained by sampling the image with a regular 2D grid of points (sampling array). The number of sampling points nX and nY in the 2D grid can correspond to the image resolution, so in the x direction, nX is the image resolution width and in the y direction, nY is the image resolution height. nX and nY are also the number of voxels in x and y directions. The voxel size (vS) is then set, for instance, based on the bounding box of the B-rep CAD model (Fig. 2). The number of voxels in the z direction, nZ, is calculated according to the volumetric model height. The volumetric representation is obtained by repeating the normalized grayscale vector nZ times (Fig. 3). Finally, the volumetric model is aligned with the B-rep CAD model.
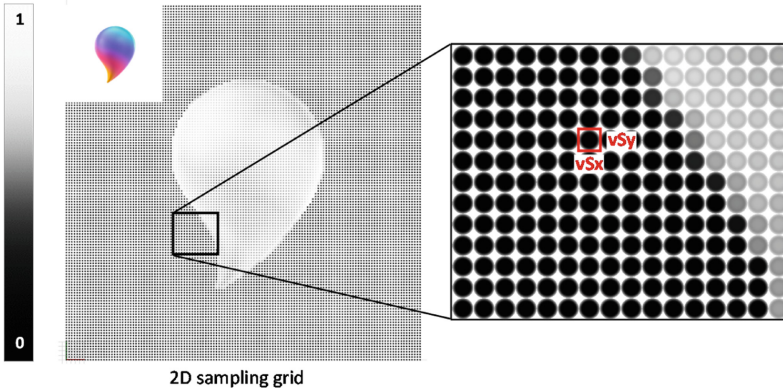
Fig. 2. The grayscale values visualized as points sampled from the color image in the top left, and a particular of the grid with voxel parameters reported (vS = vSx = vSy).
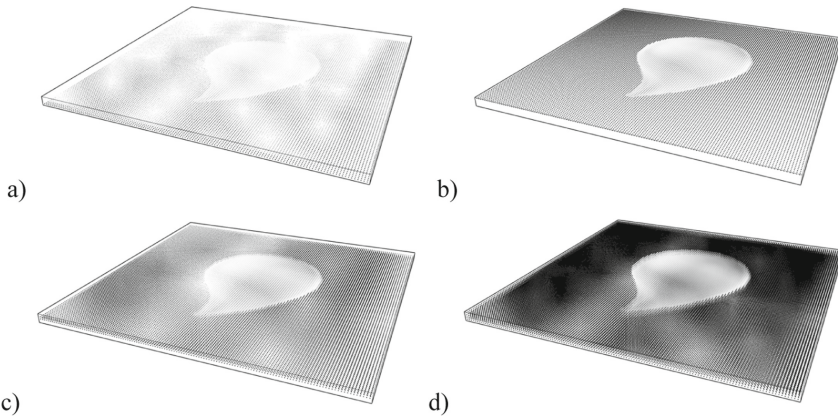

a)   b)   c)   d)

Fig. 3. 3D array visualization with nZ = 4. Single projection a) on first layer and b) on last layer; multiple projections c) 2 mid-layers and d) completely filled array (repeated nZ times).

3.3 Editing G-code

A G-code file contains the instructions as a list of lines that are read sequentially by a CNC or an FDM machine. Most of the lines in a G-code are motion instructions and are expressed by the following syntax: G1 X# Y# Z# E# F#. "X, Y, and Z" identify the cartesian coordinates of the next point to reach. "E" is the extrusion parameter; it defines the amount of filament to be fed into the extruder, moving between two consecutive points of the toolpath.

By simple consideration, it is possible to demonstrate that, for each i-th position of the toolpath in the infill portion:

$$\Delta E_{e,i} = \Delta E_{s,i} \cdot \frac{\rho_{e,i}}{\rho_s} \tag{1}$$

where $\Delta E_{e,i}$ is the amount of material needed (i.e. the value of E to substitute in the G-code during the editing phase at the i-th point of the toolpath) $\Delta E_{s,i}$ is the amount of material calculated during the slicing process, ρ_s is the infill density adopted in the slicing process and $\rho_{e,i}$ is the required relative density at the i-th point obtained by a trilinear interpolation of the volumetric model. Furthermore, in $\Delta E_{e,i}$ computation it is necessary to consider the minimum relative density ρ_{emin} allowing for a consistent part, taking into account the layer thickness. Again, this value was empirically estimated by experiments and as first attempt calculated to obtain an extrusion width E_w equal to the layer thickness.

To simplify the G-code editing, to avoid the calculation $\Delta E_{s,i}$, relative E coordinates instead of the absolute coordinates in the G-code are used. To edit the G-code a python script was implemented. Here is the pseudocode:

```
read the G-code lines
for each line:
  if available, extract X, Y, Z, E
  if line is infill:
    calculate ρe,i from the volumetric model by a trilinear interpola-
      tion on the X, Y, Z coordinates
    if ρe,i < ρemin:
      ρe,i = ρemin
    compute new_E by eq. 1
    substitute new_E in the G-code line
write the_G-code
```

4 Case Studies and Discussion

In the following section, two case studies are presented. Starting from a 2D image, it is projected into a volumetric array; thus, we can realize a 2.5D volumetric 'pattern' by repeating the same image along the Z-axis or using two or more images. In this sense, the extension to 3D is relatively simple, for instance by volumetric modeling approaches it is possible to blend different images along, for instance, the z direction. Alternatively, the volumetric model can be derived from a stack of medical images obtained from a CT scan.

4.1 Single Image

The image used has a resolution of 142×142 pixels. The bulk object is a box $80 \times 80 \times 2.4$ mm. The voxel size is 0.56 mm (isotropic). $nX = 142$, $nY = 142$ and $nZ = 4$. The grayscale values for a single layer are 20164, so 80656 values are needed to completely fill the array. The values are projected in the 4 layers of the array (Fig. 3d). The infill pattern (i.e., the lattice structure) is a gyroid with a cell size of 3.25 mm ($\rho_s = 0.35$) for an E_w range of 0.3–1.6 mm (Fig. 4a, $E_{w,min}$ = black color, $E_{w,max}$ = white color). Other process parameters are: printing temperature 215 °C, printing infill speed 25 mm/s, 3 external perimeters, no bottom and top layers; the printed object is shown in Fig. 4b, and white PLA material was adopted.
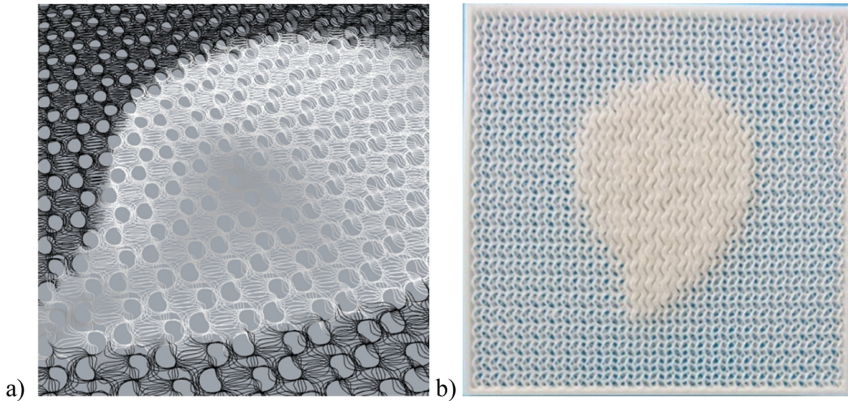
Fig. 4. Main process phases visualization: a) particular of the infill pattern lines with extrusion width values visualized ($E_{w,min}$ = black color, $E_{w,max}$ = white color); b) printed component.

4.2 Multiple Images

The two images (Fig. 5a) used in this case study have different resolutions. The "drop" image of the previous example is 142×142 pixels, whereas the symbol image (known as "Triskelion") is 232×232. The sampling array is obtained using nX = 142, nY = 142. The bulk object is a box $80 \times 80 \times 2.4$ mm. By using the resolution of the first image, the voxel size is 0.56 mm (isotropic). nZ is consequently equal to 4. The grayscale values of the drop image are projected in the 4th layer of the array, while the Triskelion values are projected on the 2nd layer, while the remaining layers are filled with 0 values The grayscale values are inverted from the previous example. The other parameters remain unchanged from the previous case study. The printed object is shown in Fig. 5b, c in which it is possible to appreciate both images, one on the top side and the other one on the bottom.
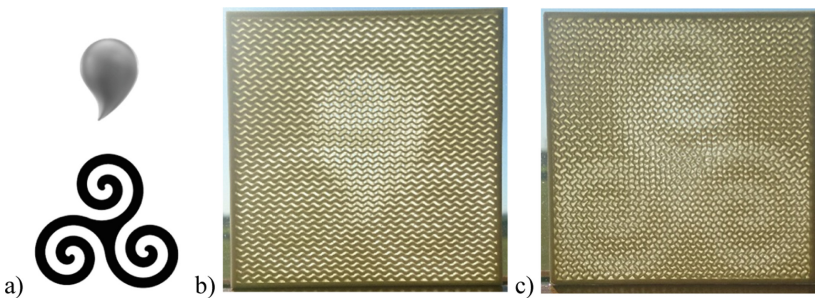

Fig. 5. a) Images used in this case study; b) printed component top side and c) bottom side.

4.3 Discussion

The proposed approach is effective in the introduction of images and embedded information in graded lattices and can be easily extended to other functional requirements, such as structural or thermal optimized infill.

The delay between the signal and the actual extrusion represents the main issue at this moment, as visible in Fig. 4b and Fig. 5b, c for adjacent paths in which the toolhead travels in opposite directions; the variation in extrusion width occurs after some length/time, causing the pattern to be out of phase. This is due to the intricate relations between process parameters and material flow [21, 22]. Even if the direct-drive configuration of the printer permits relatively low delay if compared to a bowden configuration, the problem is visible and can cause a discrepancy between the designed model and the final object.

As stated in the introductory section, avoiding the generation of a CAD model can be useful since modeling operations can be difficult even for an experienced designer. For example, boolean operations are not always performed successfully. Once the model is obtained, the slicing operations are carried out. It is worth noting that for a lattice CAD model, and especially in the case of graded lattice structures, this step is not trivial and can lead to the absence of portions of the toolpath, depending on the specific slicer software used. Based on the slicer software implementation, several attempts may be required to achieve acceptable results by carefully modifying the process parameters. Slicer software also show other limits such as the constant extrusion width, so only discrete density variations in MEX are allowed. For these reasons, in the proposed method, slicing is performed as the initial operation, therefore the generated toolpath which models the lattice structure is always consistent because it is not related to a lattice mesh model but to the process parameters. As a disadvantage, conformal lattice structures and lattices of variable size and topology cannot be obtained without other complex implementations. Moreover, since there is no CAD model, no analyses and simulations can be performed.

5 Conclusions

In this work, a new method for modeling and fabricating variable density lattices is proposed, trying to go beyond the main issues highlighted in the literature related to the design for AM and traditional CAD/CAM software.

In this approach, only a B-rep model of the object is needed. The main tasks take place directly on the G-code file as post-processing operations. By operating on the infill parameters, the lattice structure is generated and controlled. While applying a volumetric model generated from one or more images, the thickness of the infill lattice is locally tuned. The proposed method presents several advantages. First, the lattice is modeled directly as an infill pattern during slicing operations, thus the geometric model of the lattice is not needed. In such a way, the G-code is more certainly obtained, and uncoherent toolpaths are avoided. Second, the number of files and phases required from the design to fabrication of a functional lattice and the modeling efforts are lower than in standard workflows, thus avoiding some of the file exchange issues.

The proposed method can find several applications from a modeling standpoint: in artworks, embedded information on components, and functional 3D printed parts, especially in the biomedical field, e.g., the replication of the density map of a bone derived from a DICOM grayscale image to study bone-related diseases.

Future works will be aimed at testing the method on other patterns, identifying strategies allowing for conformal infill.

References

1. Tang, Y., Zhao, Y.F.: A survey of the design methods for additive manufacturing to improve functional performance (2016). https://doi.org/10.1108/RPJ-01-2015-0011
2. Nazir, A., Abate, K.M., Kumar, A., Jeng, J.-Y.: A state-of-the-art review on types, design, optimization, and additive manufacturing of cellular structures. Int. J. Adv. Manuf. Technol. **104**(9–12), 3489–3510 (2019). https://doi.org/10.1007/s00170-019-04085-3
3. Gibson, I., Rosen, D., Stucker, B., Khorasani, M.: Additive Manufacturing Technologies. Springer International Publishing, Cham (2021). https://doi.org/10.1007/978-3-030-56127-7
4. Gleadall, A.: FullControl GCode designer: Open-source software for unconstrained design in additive manufacturing. Addit. Manuf. **46**, 102109 (2021). https://doi.org/10.1016/j.addma.2021.102109
5. Giannitelli, S.M., Accoto, D., Trombetta, M., Rainer, A.: Current trends in the design of scaffolds for computer-aided tissue engineering. Acta Biomater. **10**, 580–594 (2014). https://doi.org/10.1016/j.actbio.2013.10.024
6. Savio, G., Rosso, S., Meneghello, R., Concheri, G.: Geometric modeling of cellular materials for additive manufacturing in biomedical field: A review. Appl. Bionics Biomech. 1–14 (2018). https://doi.org/10.1155/2018/1654782
7. Feng, J., Fu, J., Lin, Z., Shang, C., Li, B.: A review of the design methods of complex topology structures for 3D printing. Vis. Comput. Ind. Biomed. Art **1**(1), 1–16 (2018). https://doi.org/10.1186/s42492-018-0004-3
8. Savio, G., Meneghello, R., Concheri, G.: Geometric modeling of lattice structures for additive manufacturing. Rapid Prototyp. J. **24**, 351–360 (2018). https://doi.org/10.1108/RPJ-07-2016-0122
9. Savio, G., Meneghello, R., Concheri, G.: Design of variable thickness triply periodic surfaces for additive manufacturing. Prog. Addit. Manuf. **4**(3), 281–290 (2019). https://doi.org/10.1007/s40964-019-00073-x
10. Pasko, A., Fryazinov, O., Vilbrandt, T., Fayolle, P.-A., Adzhiev, V.: Procedural function-based modelling of volumetric microstructures. Graph. Models. **73**, 165–181 (2011). https://doi.org/10.1016/j.gmod.2011.03.001
11. Al-Ketan, O., Lee, D.-W., Rowshan, R., Abu Al-Rub, R.K.: Functionally graded and multi-morphology sheet TPMS lattices: Design, manufacturing, and mechanical properties. J. Mech. Behav. Biomed. Mater. **102**, 103520 (2020). https://doi.org/10.1016/j.jmbbm.2019.103520
12. Al-Ketan, O., Abu Al-Rub, R.K.: MSLattice: A free software for generating uniform and graded lattices based on triply periodic minimal surfaces. Mater. Des. Process. Commun. **3**, e205 (2021). https://doi.org/10.1002/mdp2.205
13. Maskery, I., Parry, L.A., Padrão, D., Hague, R.J.M., Ashcroft, I.A.: FLatt pack: A research-focussed lattice design program. Addit. Manuf. **49**, 102510 (2022). https://doi.org/10.1016/j.addma.2021.102510

14. Maltsev, E., Popov, D., Chugunov, S., Pasko, A., Akhatov, I.: An accelerated slicing algorithm for frep models. Appl. Sci. **11**, 6767 (2021). https://doi.org/10.3390/app11156767

15. Popov, D., Maltsev, E., Fryazinov, O., Pasko, A., Akhatov, I.: Efficient contouring of functionally represented objects for additive manufacturing. Comput. Des. **129**, 102917 (2020). https://doi.org/10.1016/j.cad.2020.102917

16. Gibson, L.J., Ashby, M.F.: Cellular Solids: Structure and Properties. Cambridge University Press, Cambridge (1997). https://doi.org/10.1017/CBO9781139878326

17. Moetazedian, A., Budisuharto, A.S., Silberschmidt, V.V., Gleadall, A.: CONVEX (CONtinuously Varied EXtrusion): A new scale of design for additive manufacturing. Addit. Manuf. **37**, 101576 (2021). https://doi.org/10.1016/j.addma.2020.101576

18. Holloway, A., Mamou-Mani, A., Kjelstrup-Johnson, K.: Project Silkworm. https://projectsilkworm.com/. Accessed 23 Feb 2021

19. Wheadon, M.: Velocity Painting. https://github.com/MarkWheadon/velocity-painting. Accessed 15 Dec 2021

20. Li, D., Liao, W., Dai, N., Xie, Y.M.: Comparison of mechanical properties and energy absorption of sheet-based and strut-based gyroid cellular structures with graded densities. Materials **12**, 2183 (2019). https://doi.org/10.3390/ma12132183

21. Turner, B.N., Strong, R., Gold, S.A.: A review of melt extrusion additive manufacturing processes: I. Process design and modeling. Rapid Prototyp. J. **20**, 192–204 (2014). https://doi.org/10.1108/RPJ-01-2013-0012

22. Turner, B.N., Gold, S.A.: A review of melt extrusion additive manufacturing processes: II. Materials, dimensional accuracy, and surface roughness. Rapid Prototyp. J. **21**, 250–261 (2015). https://doi.org/10.1108/RPJ-02-2013-0017