

Hierarchical, Random and Bifurcation Tiling with Heterogeneity in Micro-structures Construction via Functional Composition

Fady Massarwi^a, Jinesh Machchhar^a, Pablo Antolin^b, Gershon Elber^a

^aFaculty of Computer Science, Technion-Israel Institute of Technology, Israel

^bDepartment of Mathematics, École Polytechnique Fédérale de Lausanne, Switzerland

Abstract

This paper proposes new methods and algorithms for the construction of micro-structures that leads to a high degree of precision and water-tight models. That proposed approach involves the tiling of the domain of a deformation map using copies of a tile. The resulting tiling is functionally composed into the deformation map to obtain freeform micro-structures. Such a decoupling of the micro and macro structures allows a simplified control over design parameters. Herein, we present several extensions to this micro-structure construction scheme, comprising bivariate and trivariate tiles, implicit and parametric, including with potential discontinuities. We present support for fractal-like micro-structures, with self-similarity, allow the encoding of volumetric properties and hence heterogeneity, support for shelling and capping boundary operations, random tiling and bifurcation tiling, and support for degree reduction of the result. We verify the robustness of our approach by subjecting some of our heterogeneous micro-structures to isogeometric analysis. Finally, a variety of results from an implementation of our algorithms are also presented.

Keywords: Micro-structure, Hierarchical tiling, Random tiling, Bifurcation tiling, Functional composition.

1. Introduction

This work proposes new methods and algorithms for precise construction of micro-structures using a functional composition interface proposed recently in [14]. In that approach, the design of the macro- and the micro-structures of a porous geometry are decoupled. The micro-structure is modeled through a tile which is paved periodically in a 3D grid within the domain of a trivariate map. The macro-structure is modeled through the trivariate deformation function which maps the tiling from the parameter space to the object space. This allows for precise, water-tight construction of geometries using functional composition [11, 12].

Advances in additive manufacturing have enabled fabrication of a range of artifacts with complex and interesting properties. Use of composites, comprising several materials, has grown rapidly. The presented approach of micro-structures construction allows for the optimization of the internal structure of the material, for example, for high strength to weight ratio. This is important in many domains, for example, aviation. Random micro-structures can be used to synthesize artistic-looking geometries that can be used as 2D or 3D textures, much like 2D bump and geometric displacement mapping textures are used in computer graphics [13]. Random micro-structures were also used for designing scaffold structures in tissue engineering [2]. Another interesting application of micro-structures is in design of heat-sinks [3] where the goal is to maximize the dissipation of heat while minimizing the volume of the material used.

The computational framework in this paper is based on the one described in [14] and proposes an extended set of primitives for the construction of micro-structures. This involves functional composition of univariate, bivariate and trivariate

tiles, which are possibly trimmed, into trivariate deformation maps. Figure 1 (a) shows a tile which is paved within the domain of the trivariate map shown in Figure 1 (b) to produce the micro-structure shown in Figure 1 (c). In this example, the beams of the micro-structure in Figure 1 (c) may not be of a circular cross-section. This however, may readily be amended, if so desired, by first composing the axis curves of the beams into the trivariate map and then sweeping a circular disc along the deformed curves. Additionally, the presented computational framework introduces algorithms for generating smooth random micro-structures using implicit functions. Further, we introduce algorithms for generating micro-structures that are aware of the isometric distortion in the deformation, which enables generating uniformly sized micro-structures in Euclidean space, via bifurcations in the tiles.

At the heart of the approach in [14], lies functional composition, which, given (i) a deformation map, $\mathcal{D} : \mathbb{R}^3 \rightarrow \mathbb{R}^3$ and (ii) a tile, $T \in \mathbb{R}^3$, algebraically computes the image of T through \mathcal{D} , i.e., $\mathcal{D}(T)$. Within the scope of this work, we use either the Bézier or the B-spline representation for \mathcal{D} as well as T . We briefly describe the composition of a trivariate tile, given in Bézier or B-spline form, into a trivariate deformation map, given in Bézier form. Let the trivariate deformation map be given as:

$$\mathcal{D}(x, y, z) = \sum_{i_1=0}^{n_1} \sum_{i_2=0}^{n_2} \sum_{i_3=0}^{n_3} P_{i_1, i_2, i_3} B_{i_1}^{n_1}(x) B_{i_2}^{n_2}(y) B_{i_3}^{n_3}(z), \quad (1)$$

where P_{i_1, i_2, i_3} are the control points of \mathcal{D} and $B_{i_1}^{n_1}(x)$, $B_{i_2}^{n_2}(y)$ and $B_{i_3}^{n_3}(z)$ are the Bézier basis functions. Let the trivariate tile be given as:

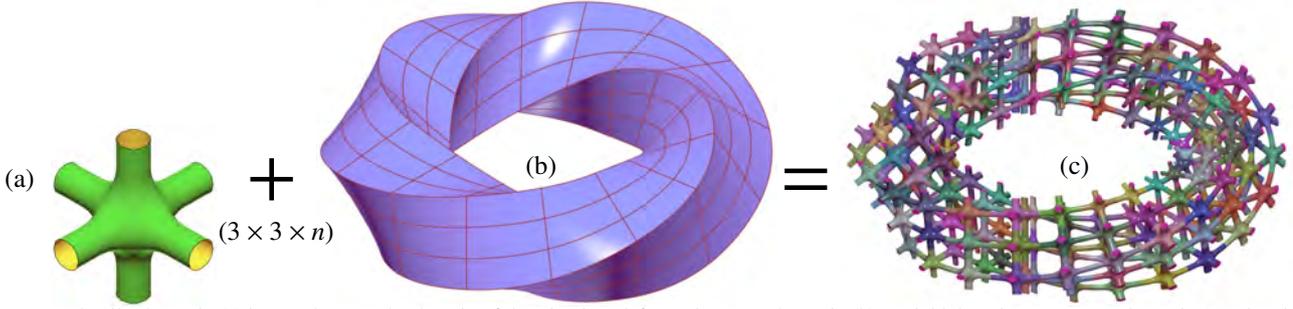


Figure 1: The tile shown in (a) is paved across the domain of the trivariate deformation map shown in (b) to yield the micro-structure shown in (c) wherein each tile is shown in a different color.

$$T(u, v, w) = \sum_{j_1=0}^{m_1} \sum_{j_2=0}^{m_2} \sum_{j_3=0}^{m_3} Q_{j_1, j_2, j_3} B_{j_1}^{m_1}(u) B_{j_2}^{m_2}(v) B_{j_3}^{m_3}(w), \quad (2)$$

where Q_{i_1, i_2, i_3} are the control points of T . The composition, $\mathcal{D}(T)$, may be expressed as:

$$\begin{aligned} \bar{T} &= \mathcal{D}(T) = \mathcal{D}(t^x(u, v, w), t^y(u, v, w), t^z(u, v, w)) \\ &= \sum_{i_1=0}^{n_1} \sum_{i_2=0}^{n_2} \sum_{i_3=0}^{n_3} \\ &P_{i_1, i_2, i_3} B_{i_1}^{n_1}(t^x(u, v, w)) B_{i_2}^{n_2}(t^y(u, v, w)) B_{i_3}^{n_3}(t^z(u, v, w)), \quad (3) \end{aligned}$$

where (t^x, t^y, t^z) are the coefficients of T . If the coefficients of control point Q_{j_1, j_2, j_3} are denoted by $(q_{j_1, j_2, j_3}^x, q_{j_1, j_2, j_3}^y, q_{j_1, j_2, j_3}^z)$, then the composition $B_{i_1}^{n_1}(t^x(u, v, w))$ may be computed as:

$$\begin{aligned} B_{i_1}^{n_1}(t^x(u, v, w)) &= \binom{n_1}{i_1} t^x(u, v, w)^{i_1} (1 - t^x(u, v, w))^{n_1 - i_1} \\ &= \binom{n_1}{i_1} \left(\sum_{j_1=0}^{m_1} \sum_{j_2=0}^{m_2} \sum_{j_3=0}^{m_3} q_{j_1, j_2, j_3}^x B_J^M(u, v, w) \right)^{i_1} \\ &\quad \left(1 - \sum_{j_1=0}^{m_1} \sum_{j_2=0}^{m_2} \sum_{j_3=0}^{m_3} q_{j_1, j_2, j_3}^x B_J^M(u, v, w) \right)^{n_1 - i_1}, \quad (4) \end{aligned}$$

where $B_J^M(u, v, w) = B_{j_1}^{m_1}(u) B_{j_2}^{m_2}(v) B_{j_3}^{m_3}(w)$. The functional compositions for $B_{i_2}^{n_2}(t^y)$ and $B_{i_3}^{n_3}(t^z)$ are performed similarly. The product of Bézier or B-spline basis functions in Equation (4) can be performed precisely [12, 17, 18]. The functional composition of univariate and bivariate geometries into trivariate tiles can be done in a similar way. For further details on functional composition of tiles, refer to [14].

In the framework of [14], a tile is given along with a deformation map. Beyond the obvious requirement that the tiling must be contained within the domain of the deformation map, the computational framework allows for tiles and deformation maps of arbitrary dimensions. For the convenience of demonstration though, we assume the deformation map to be a trivariate in this paper.

The rest of this paper is organized as follows. In Section 2, we summarize the previous approaches towards constructions of micro-structures. An extended set of methods is proposed for the construction of hierarchical micro-structures, in Section 3. In Section 4, a method for generating random micro-structures is presented, and in Section 5, a new algorithm for generating micro-structures with bifurcation tiling is presented. Section 6

discusses more extensions such as shelling and capping of the boundaries of the constructed models, representations of heterogeneous micro-structures, and lower order approximations of the created micro-structures which can be of high degree due to the functional composition operation. As a testimony to the robustness of the proposed interface, Section 7 discusses isogeometric analysis over heterogeneous micro-structures constructed using the presented approaches. A variety of results from the proposed framework are presented in Section 8. Finally, the paper is concluded, in Section 9, with comments on directions along which this work may be extended.

2. Previous work

The computer aided design approach towards the creation of micro-structures is a relatively nascent area, mostly due to the difficulty in manufacturing such shapes. Recent advances in additive manufacturing [19] have opened new avenues in this direction. We broadly classify micro-structures and their design methodologies into grid-based and random. Here we mostly focus on the grid-based approaches, these being related to our proposed approach.

The authors in [4] perform optimization of micro-structures towards multiscale modeling. A set of parameters is identified at the microscale which govern properties such as the shape, stress, strain, etc. of the model. The finite element method is used for the analysis step. Evolutionary schemes, which are based on genetic algorithms, are used to search for optimal values of parameters. The NURBS representation is used for the shape of the structures, but such an approach lacks precision.

In [8], the authors present a framework for modeling heterogeneous objects using trivariate Bézier patches. The Bézier patches have two sets of coordinates. The first set of three coordinates, (x, y, z) , prescribe the shape of the object, while the rest of the coordinates specify the material composition of the object. While this allows for construction of a wide range of structures with varied materials, it only admits a single level of details, and is limited to Bézier trivariates only. Modeling of micro-structures, for instance, is beyond the scope of this work.

The authors in [32] adopt principles from stochastic geometry [21] for designing porous artifacts. While such an approach is suitable for designing random micro-structures, for example the bone tissue, it is less favorable for applications requiring a high degree of precision, for instance, a wing of an aircraft. Also, the approach doesn't address connectivity of the porous structures.

The authors in [35] use Voronoi tessellations to generate random porous structures of three kinds: porous geometries with

intersecting fractures, interconnected tubes and fibers. A set of points is randomly sampled in the space which gives rise to Voronoi tessellations. Offsets are then computed for the edges of the tessellations to generate the three kinds of micro-structures. This approach is limited to constructing geometry which is piecewise linear or cylindrical.

In [28], the authors use implicit surfaces for modeling micro-structures. Their approach allows for design of regular as well as irregular structures which are amenable to geometric operations such as blending and deformation. Such an approach does not readily support creation of heterogeneous media. Moreover, the extension to parametric representation is not directly supported. Additionally, their approach doesn't guarantee connectivity for the case of irregular (random) structures.

In [2], the authors propose a method for creating scaffolds as support structures for tissue engineering. The scaffolds, once fabricated out of some biodegradable or bioresorbable material, are seeded with (biological) cells and provide support and shape to tissue during its growth. In this approach, the scaffolds are modeled as porous micro-structures using polygonal representation. The method does not generalize to freeform spline geometry and heterogeneous materials are not supported.

In [33], the authors propose a method for designing mesoscopic structures using trusses. The output of the system is in the form of triangles in STL format. Since this method is tailored to trusses as the basic building blocks of the structure, it has limited scope of application. The authors mention tiling the volume of trivariates with tiles, though no details are provided.

The authors in [6] propose a framework for design towards additive manufacturing of mesostructures. Their method is based upon the process-structure-property-behavior model. The basic building element is an octet truss which is represented parametrically. Extension for support of other types of elements is lacking.

In [25], the authors propose a method to design cellular structures geared towards additive manufacturing. The cellular structure is achieved through an adaptive triangulation of the interior of the solid, with finer tetrahedra along the boundary of the solid. Their method also supports a dual construction obtained from the Voronoi diagram of the triangulation. The approach does not generalize to freeform geometry.

In [22], the authors propose a modeling primitive based on a generalized cuboid shape, which is referred to as a block. The design of a complex object proceeds by laying out blocks, which are then connected to form the basic shape of the object to be modeled. A control mesh is then extracted from the faces of the blocks, which allows parameterization of the surface. While this provides for a simple and elegant modeling approach, it does not allow the two stage methodology for the design of micro and macro structures.

To summarize, the previous approaches suffer from one or more of the following shortcomings, which are addressed by the work presented here: support for freeform parametric geometry, support for bifurcations in the tiling, compensation for the non-isometric behavior of the deformation map, support for heterogeneous materials and hierarchical structures, and a high degree of precision.

The construction of micro-structures involves two main steps: (i) Placement of tiles within the domain of the deformation map to achieve a hierarchical tiling and (ii) Composing the tiling created in step (i) into the deformation map to obtain the required micro-structure. Step (ii) was described in Section 1

and is not new. In the coming sections, we describe an extended set of primitives towards step (i), which lead to a whole new variety of micro-structures.

3. Hierarchical Micro-structures

In order to create hierarchical micro-structures, the tile is no longer placed in a three dimensional regular grid. Instead, a **branch-in** face, and (possibly more than one) **branch-out** face(s) must be specified on the tile. If the tile has more than one branch-out face, then the tile is said to have a bifurcation [31]. Further, the branch-in and branch-out faces must be mappable. In other words, the branch-out face(s) must be similar to the branch-in face, up to some (simple) mapping, i.e., $M(\text{branch-out}) = \text{branch-in}$ for some simple map M . Two approaches are proposed towards these hierarchical constructions. Section 3.1 discusses hierarchical micro-structure's construction using trivariate tiles with discontinuities, wherein the hierarchical information is implicitly encoded within the discontinuities of the tile. Section 3.2 achieves the same using fractal-like trivariate tiles, wherein the hierarchical information is explicitly encoded within the tile.

3.1. Hierarchical micro-structures using tiles with discontinuities

One way to achieve bifurcation is through partial Euclidean discontinuities that are introduced into the trivariate tiles. A trivariate T is said to have a partial Euclidean discontinuity, say along the w parameter direction, if there exists w_1 and w_2 such that the surface $T(u, v, w_1)$ is continuous while the surface $T(u, v, w_2)$ is not. See Figure 2. A partial Euclidean discontinuity leads to a geometric split within the tile, which in turn leads to bifurcations. For example, the four surfaces $S_i, i = 0, \dots, 3$, in Figure 2 (a), two of which with discontinuities, can be skinned to form the tile T in Figure 2 (b). The branch-in face of T is its bottom rectangular base and its branch-out face(s) are the six top split rectangular regions. A recursive placement of tiles so that the branch-in bases of the tiles in the next level are mapped to the branch-out face(s) of the current level, completes the definition of the hierarchical construction. Figure 2 (c) shows three such recursive levels.

Thus, a tile with n discontinuities along any direction has $n + 1$ branches along that direction. For the ease of discussion, we assume that the discontinuities in the trivariate tiles are only along u and/or v directions. Hence, a tile with n and m discontinuities along the u and v directions, respectively, will have, in all, $(n + 1)(m + 1)$ branches. For instance, the tile shown in Figure 2 (b) has two discontinuities along the u direction and one along v , and hence, six branch-out faces at the top.

Recall that the branch-in face and the branch-out faces are to be related by some map. Herein, an affine transform is employed which maps the branch-in face to each of the branch-out faces. For simplicity, here we restrict the branch-in and the branch-out faces to be planar. A tile with n branch-out faces will have n such affine transforms.

3.2. Hierarchical micro-structures using fractal-like tiles

A different approach to designing hierarchical micro-structures arises from explicit labeling of bifurcations within the tile. In this scheme, the tile is made of one or more trivariates. The user labels one of the six boundary faces of one of the

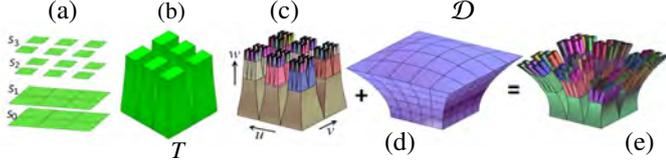


Figure 2: The trivariate tile, T , in (b), with two discontinuities along u and one along v , was constructed by skinning through the four surfaces in (a), two of which (S_2 and S_3) have discontinuities. The skinning operation proceeds by using the control-mesh of the surfaces as the control-mesh of the resulting trivariate. Hence the intermediate surfaces (S_1 and S_2 here) are not interpolated. A recursive hierarchy, in the domain of the deformation map is built using T and is shown in (c), with three levels. The tiles in (c) are composed into the trivariate map, \mathcal{D} , shown in (d), yielding (e). All tiles in (c) and (e) are shown in different colors. The degrees of T , \mathcal{D} and $\mathcal{D}(T)$ in (u, v, w) are $(2, 2, 3)$, $(2, 2, 2)$ and $(12, 12, 18)$, respectively.

trivariates, as a branch-in face. In addition, the user labels one or more boundary faces of the trivariates as branch-out faces.

In each recursive pavement of tiles, for each face labeled as a branch-out face, a mapped version of the tile is placed over the original tile so that the face labeled as the branch-in face coincides with the one labeled branch-out. Again, here we restrict the branch-in and branch-out face to be related by some affine transform. An example of such a tile is shown in Figure 3 (a) which has three trivariate elements. The face labeled S_0 is designated as the branch-in face while those labeled S_1, S_2 are designated as branch-out faces. The tile is paved recursively with 7 levels, as shown in Figure 3 (b) and then composed into the deformation map shown in Figure 3 (c) to obtain the model shown in Figure 3 (d).

We now highlight the merits of the methods proposed in Sections 3.1 and 3.2. While they are related and it may seem that the method of Section 3.1 can be implemented via that of Section 3.2, this would require a way to aggregate all $(n+1)(m+1)$ bottom faces of the tile into a single face, before computing the affine maps between the branch-in and branch-out faces of a trivariate tile, having n and m discontinuities along u and v directions, respectively.

4. Random Implicit Micro-structures

The methods described in previous sections enable the modeling of micro-structures based on repetition and deformation of a single tile into a regular or hierarchical grid that is composed into a deformation map. In order to guarantee connectivity and smoothness, the boundaries of the tile should be designed to comply with some C^k connectivity requirements. In this Section, we propose a method for designing irregular randomized micro-structures that are C^1 continuous.

Having a 3D grid that consists of m_x, m_y, m_z cells in the x, y, z directions respectively, we build a random tile in each cell, while taking care of the connectivity and smoothness between the cells, as following: For each cell C_{qrs} , $0 \leq q < m_x, 0 \leq r < m_y, 0 \leq s < m_z$, a scalar tensor product trivariate B-spline function $f^{qrs}(x, y, z)$ is defined:

$$f^{qrs}(x, y, z) = \sum_{i=0}^{n_x} \sum_{j=0}^{n_y} \sum_{k=0}^{n_z} p_{i,j,k} B_{i,d_x}(x) B_{j,d_y}(y) B_{k,d_z}(z), \quad (5)$$

where f^{qrs} is defined over the box domain of C_{qrs} , and where $p_{i,j,k} \in [-1, 1], 0 \leq i \leq n_x, 0 \leq j \leq n_y, 0 \leq k \leq n_z$, are the

control coefficients of f^{qrs} and $B_{i,d}$ is the i 'th univariate B-spline basis functions of degree d , with an open end knot sequence.

The tile in cell C_{qrs} is defined as the implicit function $f^{qrs}(x, y, z) = 0$. Herein, f^{qrs} is positive inside the tile and negative outside the tile.

We consider the following design parameters that are given by the user:

- The degrees, d_x, d_y, d_z , and number of control coefficients, n_x, n_y, n_z for f^{qrs} in each direction for every cell.
- A canonical trivariate f_0^{qrs} , that is in the same function space as f^{qrs} .
- A randomization factor, $\alpha \in [0, 1]$, that controls the randomness of f^{qrs} relative to f_0^{qrs} . That is, for $\alpha = 0$, $f^{qrs} = f_0^{qrs}$, and for $\alpha = 1$, the coefficients, $p_{i,j,k}$, of f^{qrs} are completely random.

Then, we define the coefficient $p_{i,j,k}$ of f^{qrs} as the following:

$$p_{i,j,k} = (1 - \alpha)p_{i,j,k}^0 + \alpha \text{Random}(-1, 1), \quad (6)$$

where $p_{i,j,k}^0$ are the coefficients of f_0^{qrs} and $\text{Random}(-1, 1)$ is a random number in the domain $[-1, 1]$.

Finally, we employ the marching cubes algorithm [23] to extract a zero set approximation of the geometry of the implicit function $f^{qrs}(x, y, z) = 0$.

4.1. C^1 smoothness

Defining a C^1 continuous trivariate f^{qrs} will result in a C^1 smooth tile, in general. However, smoothness between tiles should also be addressed. In each boundary of the cell, C^0 continuity can also be guaranteed by making sure the trivariates in neighboring cells share similar control coefficients along their common boundary, and C^1 continuity can be achieved by placing the first inner layer of control coefficients from both sides of the common boundary at the same displacement from the shared control coefficients on the boundary between the cells. Consider two cells $c_{q,r,s}$ and $c_{q+1,r,s}$ having control coefficients p_{ijk} and q_{ijk} , respectively. Then, C^0 continuity implies (for the yz boundary face):

$$p_{n_x,j,k} = q_{0,j,k}, \quad 0 \leq j \leq n_y, 0 \leq k \leq n_z, \quad (7)$$

and for C^1 continuity, in addition to Equation (7), we require:

$$(p_{n_x,j,k} - p_{n_x-1,j,k}) = (q_{1,j,k} - q_{0,j,k}), \quad 0 \leq j \leq n_y, 0 \leq k \leq n_z. \quad (8)$$

Note that due to the grid topology, internal junctions are regular (valence 8).

4.2. Global connectivity

The construction of the f^{qrs} trivariates result in C^1 implicit tiles. However, the tiles might form several disconnected components, globally, which is likely to be undesired. In order to have one globally connected component, we enforce local (Intra-Cell) connectivity, in Section 4.2.2, i.e each tile must be a simple connected component as well as global (Inter-Cell) connectivity, in Section 4.2.1, by guaranteeing connectivity between all tiles.

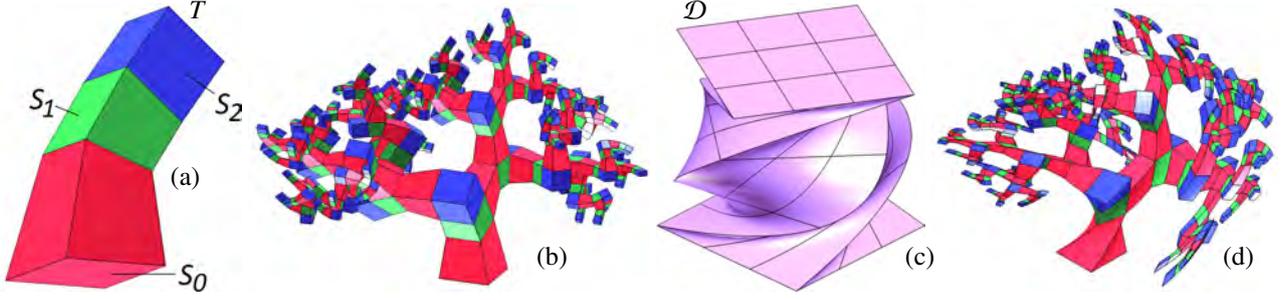


Figure 3: The tile T , shown in (a) has three trivariate elements. The boundary face labeled S_0 is designated as branch-in face and the ones labeled S_1 and S_2 are designated as branch-out faces. The tile is paved recursively, with 7 levels, as shown in (b) and then composed into the deformation map, \mathcal{D} , shown in (c) to obtain a hierarchical micro-structure, $\mathcal{D}(T)$, shown in (d). The degrees of T , \mathcal{D} and $\mathcal{D}(T)$ in (u, v, w) are $(1, 1, 1)$, $(3, 3, 2)$ and $(8, 8, 8)$, respectively.

4.2.1. Inter-cell connectivity

We now introduce an algorithm for ensuring a topological face connectivity between all cells. First, we construct a graph G consisting of the centers of the cells as its vertices, and all (up to six) edges that connect the centers of neighboring cells that share a face. Then, a spanning tree [9], denoted L_1 , consists of edges in G that connects all the cells is constructed. This can be done by picking a seed edge in G and performing a depth first search until all the cells are visited. A second spanning tree L_2 , is optionally constructed in the same way after removing all the edges of L_1 from G . Hence, L_1 and L_2 have no edge in common. Since each tree, L_i , that passes through a cell intersects two (or more) faces of the cell (except for leaf cells in L_i), and since each cell have six faces, the maximal possible number of such trees is three. The existence of L_2 depends on L_1 , and it is not always possible to find an L_2 . See for example L_1 in the grid in Figure 4 (a), for which no L_2 exists. Yet, practically, cases for which no L_2 was found are rare. Note that there could be at most three disjoint spanning trees. Further, it is the user's choice to use one, two or three trees, if exists.

The connectivity graph $L = L_1 \cup L_2$ is considered as all the edges in L_1 (and L_2). If L_2 is provided, the graph L ensures connectivity of two, which means that each two cells are connected with two separate paths, optionally improving the stability of the result. Each edge $e \in L$, represents a boundary face b shared between two neighboring cells. Consider cells $C_{q,r,s}$ and $C_{q+1,r,s}$, having implicit trivariates $f_{q,r,s}$ and $f_{q+1,r,s}$, respectively, with coefficients $p_{i,j,k}$ and $q_{i,j,k}$, and share boundary face b (i.e Equation (7)). Since the trivariates are B-spline functions with an open end condition, the values of the trivariates on boundary b depend only on the control coefficients that lie on b , namely $p_{n_x,j,k}$ ($= q_{0,j,k}$). Hence, the zero set restricted to b , is the (implicit) curve connecting cells $C_{q,r,s}$ and $C_{q+1,r,s}$, and depends only on these coefficients. To make sure the zero set is not empty on face b , the coefficients $\{p_{n_x,j,k}\}$ are initialized to negative values, and an iterative process that increases the number of positive coefficients from $\{p_{n_x,j,k}\}$ on even iterations, and in odd iterations, the values of $\{p_{n_x,j,k}\}$ are increased until the zero set is no longer empty on b . Clearly, other strategies of increasing the positivity of the coefficients can be employed. Detecting a non-empty b is accomplished by sampling b until both positive and negative samples are detected. $\{q_{0,j,k}\}$ are assigned the same values computed for $\{p_{n_x,j,k}\}$. Figure 4 shows a connectivity tree L_1 and the constructing tiles, accordingly. Figure 5 (c) shows a connectivity graph (consists of two trees L_1 and L_2) which is used in the construction of a random micro-structure of the duck model in Figure 14.

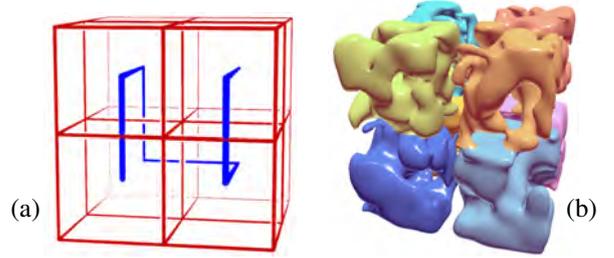


Figure 4: The connectivity of the (eight) tiles follows spanning tree L_1 , in blue in (a). Tiling of eight random tiles with ensured intra- and inter-cell connectivity is presented in (b), following L_1 . No spanning tree L_2 exists for this case.

4.2.2. Intra-Cell connectivity

In Section 4.2.1, we showed how to ensure that the zero set of the implicit trivariates is properly shared with its neighbors. We now show how one can control the internal coefficients of the trivariate, in order to guarantee the zero set inside the cell is forming one connected component. By repeatedly increasing the positive valued internal coefficients (except for coefficients of first layers in each direction that ensure C^1) as described in the iterative process in Section 4.2.1, the components of the zero set inside the cell must connect together into one component. As in the limit when all internal coefficients are positive, the entire cell is inside, up to the boundary conditions. After each such iteration, the zero set is extracted and examined, and the process continues until the zero set in the cell is found to form a one connected component.

5. Adaptive Bifurcation Micro-structures

Using a regular grid, introducing the same number of cells in each layer, can have a major drawback: the tiles have a uniform size in the domain of the deformation map, \mathcal{D} , which means that after the mapping, their sizes might vary significantly, due to non-isometric behavior of \mathcal{D} .

It might be desired to form tiles that preserve, as much as possible, their sizes in the Euclidean space. In this section, we present an algorithm that constructs tiles of approximately similar size in Euclidean space. By using non-regular adaptive subdivision, a non-grid topology can be formed, in the domain of the deformation map. In a regular grid tiling in the parametric space, each tile has at most one neighboring tile in each direction. Herein, bifurcations in tiles are possible, forming the non-grid topology.

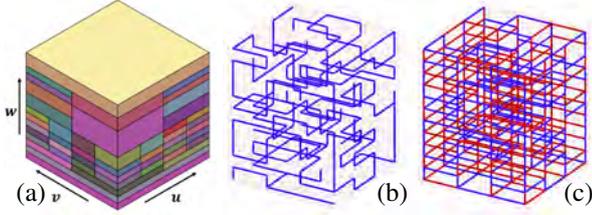


Figure 5: In (a), adaptive domain subdivision of the duck trivariate from Figures 15 and 16, according to Algorithm 1, with $\epsilon = 0.35$ (where the length of the duck is little over 2). The w direction starts from the tail and ends with the head of the duck. (b) shows the L_1 connectivity graph (in blue), where as (c) shows the L_2 connectivity graph (in red), over L_1 . The connectivity graph in (c) is used in the construction of the random micro-structure in Figure 14.

Given a deformation map \mathcal{D} , the topology of the tiling is adaptively built by subdividing the parametric space of \mathcal{D} until each cell satisfies a certain condition on \mathcal{D} . The determinant of the Jacobian of \mathcal{D} , $|J| = \langle \frac{\partial \mathcal{D}}{\partial u} \times \frac{\partial \mathcal{D}}{\partial v}, \frac{\partial \mathcal{D}}{\partial w} \rangle$, might be used as such a classification condition. However, $|J|$ does not hint on the preferred direction to subdivide. Alternatively, in this work, we use the following subdivision strategy: Given a sub-domain, D , of \mathcal{D} , and an Euclidean space subdivision threshold ϵ , compute the image of D 's eight corner points under \mathcal{D} . Let e be the edge of D with the maximal length under \mathcal{D} 's image. If $|e| > \epsilon$, we subdivide D at the middle of its parametric domain, in the direction of e . Otherwise, the subdivision process terminates. This adaptive topological bifurcation is described in Algorithm 1 and is initialized with the call $TopoSubdiv(\mathcal{D}, D, \epsilon)$. Figure 5 (a) shows an example of an adaptive topological bifurcation used to generate the duck models in Figures 15 and 16 (b and c), with $\epsilon = 0.35$, where the length of the duck is little over 2.

Algorithm 1 TopoSubdiv: Topological subdivision of \mathcal{D} 's domain

Input:

\mathcal{D} : a deformation map;
 D : sub-domain of \mathcal{D} to subdivide;
 ϵ : a subdivision threshold;

Output: a topological tiling of the domain of \mathcal{D} ;

Algorithm:

- 1: $\mathcal{E} :=$ twelve edges of D 's box;
 - 2: $e_{max} :=$ edge in \mathcal{E} with maximal length under \mathcal{D} 's image;
 - 3: **if** $\|e_{max}\| < \epsilon$ **then**
 - 4: **return** D ;
 - 5: **end if**
 - 6: $axis_e :=$ parametric direction of e_{max} ;
 - 7: $\{D_L, D_R\} :=$ Subdivide D in the middle, along $axis_e$;
 - 8: **return** $TopoSubdiv(\mathcal{D}, D_L, \epsilon) \cup TopoSubdiv(\mathcal{D}, D_R, \epsilon)$;
-

This adaptive topological subdivision is used to construct a tiling with bifurcation. In Section 5.1, a tiling method for generating bifurcation for implicit micro-structures is presented, and in Section 5.2 bifurcation for parametric micro-structures is introduced.

5.1. Bifurcation for implicit micro-structures

The implicit random tiles' synthesis, that was presented in Section 4, simplifies the automated generation of bifurcation

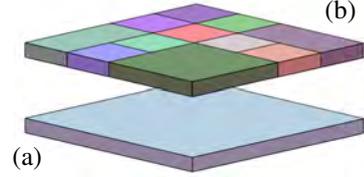


Figure 6: A complex face in (a) with its neighboring faces in (b). The faces in (b) are stitched recursively along their shared edges to construct one B-spline surface with the same size of the face in (a).

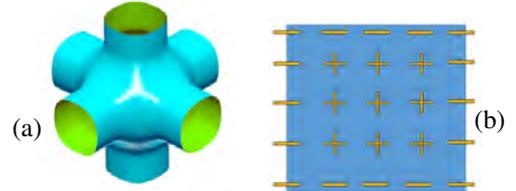


Figure 7: A tile in (a) is constructed as the zero set of a volumetric Boolean sum [16] of six B-spline faces having coefficients pattern as in (b). The same construction method of the tile in (a) is used to construct the tiles of the canonical model of the duck in Figure 16 (b)

tiles. Consider a cell C in a topological subdivision G created using Algorithm 1. Every face of C can be topologically classified as one of the following:

1. A boundary face of G .
2. Identical to a neighbor cell face. This is the case where at this face there is only one neighboring tile.
3. A complex of several faces. This is the case where bifurcation is performed along the direction of this face.

If all the faces of C are of the first or second type, then C is considered a simple cell and the trivariate defining its tile is constructed as described in Section 4. If C has at least one complex face of the third type, then the cell is considered as a bifurcation cell. Building the implicit trivariate for a bifurcation cell is done as following:

1. Build 6 boundary B-spline surfaces, one surface for each face. For a simple face (type 1 or 2 above) the coefficients of the trivariate on that face are assigned random values as in Equation (6). For a complex face (type 3), merge all the neighboring surfaces (that are already constructed) into one B-spline surface. See an example of a highly complex face in Figure 6.
2. Construct a trivariate, using volumetric Boolean sum [16], over these 6 faces from Step 1. See, for example, the tile in Figure 7 (a) constructed as the zero set of the Boolean sum trivariate of six B-spline surfaces having the same structure of coefficients as shown in Figure 7 (b), having negative values on the boundaries and positive values in the interior.
3. Randomize the internal coefficients of the trivariate from Step 2 according to randomization factor α as described in Equation (6). C^1 connectivity between the tiles is ensured as described in Section 4.1.

5.2. Bifurcation for parametric micro-structures

In the case of parametric tiles, there is no simple automated method for creating the bifurcation tiles for arbitrary general

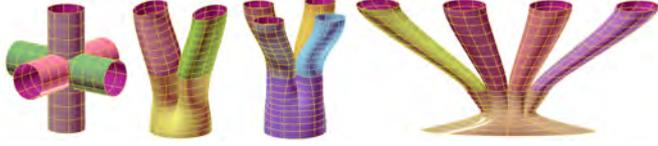


Figure 8: Different tiles formed out of parametric B-spline surfaces, to be employed in bifurcations in micro-structures (See also Figure 15). Each tile is composed of several bi-cubic B-spline surfaces, each surface is highlighted with a different color.

topology. Hence, tiles with proper connectivity and continuity conditions are a-priori defined. The input bifurcation tiles are classified according to their bifurcation topology. For instance, 1 to 2 connection, 1 to 4, etc. Based on the topology of the bifurcation tiles provided, the divided structure in Algorithm 1 is restricted to subdivisions that are confirmed with the prescribed bifurcation tiles. Herein, we restrict the bifurcation to be along layers in one direction, and allow only three types of bifurcations: 1 to 2 and two types of 1 to 4. These tiles, formed out of tensor product B-spline surfaces, are presented in Figure 8.

6. Additional extensions

In this section, we describe additional functionalities which aid the design and fabrication of micro-structures. Section 6.1 discusses appending the geometric description of trivariate micro-structures with material or properties description. This, for example, allows the design of micro-structures with heterogeneous material composition. Section 6.2 discusses operations on micro-structure along the boundary of the deformation map which ensure that the resulting model is sealed. Finally, Section 6.3 presents a simple optional degree-reduction scheme for the resulting composed freeforms, that can be of high degrees.

6.1. Heterogeneous micro-structures

Advances in additive manufacturing have enabled fabrication of artifacts with non-trivial material composition. The modeling of heterogeneous micro-structures composed of trivariate tiles is made possible by encoding the information about the material in the parameter space of the geometry. While this step is similar to [8], the functional composition widens the scope of our approach. The material properties are supported by an additional set of coordinates of the trivariate tile. Such a trivariate tile is a vector function $(x, y, z, c_1, \dots, c_n)$, where x, y, z, c_1, \dots, c_n are the coefficients of the trivariate. The first three coordinates encode the geometry of the tile while the last n coordinates encode (material) properties. One can set the (material) properties, i.e., c_1, \dots, c_n , before as well as after the composition of the tile into the deformation map. Properties assigned to some geometry in the tile, a surface or a trivariate, are propagated along unaltered if the tile undergoes composition. This gives the flexibility to the user to control the manner in which the deformation affects the material properties, if at all.

6.2. Shelling and capping operations

In order to produce a water-tight B-rep micro-structure, tiles must be sealed along the boundaries of the deformation map, \mathcal{D} . The pre-image of the micro-structure along the six boundary surfaces of the domain of \mathcal{D} can be closed via two alternative approaches:

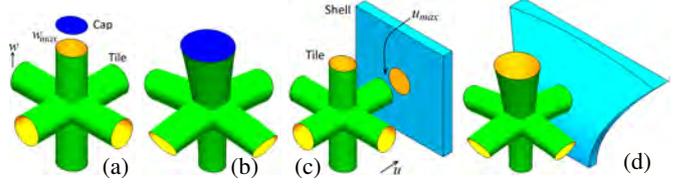


Figure 9: Illustration of shelling and capping operations. A tile consisting of three open cylindrical surfaces and a cap (in blue) along the $w = w_{max}$ surface, are shown separately, in (a), before applying the deformation map. The tile and the cap are composed into the deformation map shown in Figure 2 (d) and the resulting model is shown in (b). The tile and a shell (in cyan) element along u_{max} before deformation are shown separately in (c). (d) shows the tile and the shell after composing into the deformation map shown in Figure 2 (d).

1. **Capping:** The first approach involves capping of all the open holes along the boundary surfaces of the deformation map, \mathcal{D} . For ease of discussion and without loss of generality, we assume that the axis-aligned bounding box of the tile is the unit cube. Boundary curves are detected and extracted from the tile, if on one of the six planes of $x = 0, x = 1, y = 0, y = 1, z = 0, z = 1$, as a pre-processing step. If a copy of the tile is placed along the $u = u_{min}$ surface of the domain of \mathcal{D} , then the boundary curves of the tile for $x = 0$, if any, are used to create a planar trimmed surface which serves as a cap over the tile along that $u = u_{min}$ surface. These capping surfaces are becoming part of the tile and undergo deformation with the tile. A similar process is employed for tiles placed along all boundaries of \mathcal{D} . Figure 9 (a) and (b) show a simple capping example along the $w = w_{max}$ surface.

For implicit tiling, capping can be achieved by assigning negative values to all coefficients of a face that lie on the boundaries of \mathcal{D} . The distance of the zero set from the boundary of \mathcal{D} can be controlled by the value of the control points on the boundary:

Lemma 1. Consider a scalar Bézier curve $C(t) = \sum_{i=0}^n p_i \theta_{i,n}(t)$, and assume $|p_i| \leq 1$ for $1 \leq i \leq n$. If $C(t_0) = 0$, for some t_0 , then it holds that $t_0 \geq 1 - \sqrt[n]{\frac{1}{1-p_0}}$.

Proof:

Because $0 = C(t_0) = \sum_{i=0}^n p_i \theta_{i,n}(t_0)$, we have,

$$p_0(1 - t_0)^n = \sum_{i=1}^n -p_i \theta_{i,n}(t_0) \geq \sum_{i=1}^n -\theta_{i,n}(t_0) = (1 - t_0)^n - 1.$$

In other words, $(1 - p_0)(1 - t_0)^n \leq 1$ and $t_0 \geq 1 - \sqrt[n]{\frac{1}{1-p_0}}$. ■

Similar behavior to the one pointed by Lemma 1, can be expected from B-spline curves with open end conditions. Now, consider the coefficients of trivariates f^{qrs} of a cell that is on the boundary of \mathcal{D} . Arrange the order of the summations in Equation (5), such that the inner most summation is orthogonal to the boundary. Assume w.l.o.g. that

this direction is the z direction. Then:

$$\begin{aligned} f^{qrs}(x, y, z) &= \sum_{i=0}^{n_x} \sum_{j=0}^{n_y} \sum_{k=0}^{n_z} p_{i,j,k} B_{i,d_x}(x) B_{j,d_y}(y) B_{k,d_z}(z) \\ &= \sum_{i=0}^{n_x} \sum_{j=0}^{n_y} c_{ij}(z) B_{i,d_x}(x) B_{j,d_y}(y), \end{aligned}$$

where,

$$c_{ij}(z) = \sum_{k=0}^{n_z} p_{i,j,k} B_{k,d_z}(z).$$

Hence, using Lemma 1 and seeing f^{qrs} as $(n_x + 1)(n_y + 1)$ curves, one can control the distance between the zero set of $f^{qrs}(x, y, z)$ and the boundary of \mathcal{D} . Lemma 1 provides a lower bound of the distance (ranging from 0 to 1) of the zero set from the boundary. If $p_{i,j,0} = 0$ the zero set is on the boundary (assuming $p_{i,j,k} \neq 0, \forall k > 1$), and as $p_{i,j,0}$ gets smaller (having a negative value), the zero set moves inside.

2. **Shelling:** The second approach involves creating a complete shell of some thickness, as specified by the user. As in the capping approach, a preprocessing step extracts the boundary curves of the tile. The construction of the shell for a tile starts with the creation of two planar surfaces, for the outer and inner shells, with opposite orientations. The inner surface is trimmed using the appropriate boundary curves of the tile, but with reversed orientation. The outer and the inner surfaces are then joint using four planar surfaces to form a closed (trimmed) box. This is illustrated by a simple example in Figure 9 (c) and (d). For implicit tiling, shelling can be achieved by assigning positive values to all coefficients of faces on the boundary of \mathcal{D} , and adding the original boundary surfaces of \mathcal{D} .

6.3. Lower order approximations

The functional composition operation of parametric forms described in the previous sections can lead to geometries with high degrees. In addition to the large space complexity, this can lead to higher computational costs in downstream operations. In order to alleviate this issue, we also support simple lower degree approximation of micro-structures by quadratics and cubics, while preserving continuity between glued patches.

This simple approximation goes as follows: B-spline entities in the constructed micro-structures are first divided into Bézier entities. Bézier curves in the composed tiles are then approximated by either quadratic or cubic Bézier curves using an approximation that ensures interpolation of the end points and reconstruction of tangents at the end points (in the cubic case). Bézier surfaces are then reconstructed from four boundary Bézier curves, possibly exploiting the interior surface control point(s) to reduce approximation error in a least squares sense. Going to the next level, Bézier trivariates are then reconstructed from six boundary Bézier surfaces, possibly exploiting the interior trivariate control point(s) to reduce approximation error in least squares sense. Consider now two adjacent high order freeforms sharing a common face F of a lower dimension. Both freeforms are approximated using the same approximation procedure just described. The boundary of an approximated (Bézier) freeform, denoted $A(F)$, is only affected by the input shared boundary, F . Hence, both approximated freeforms will now share a common face $A(F)$, ensuring continuity. Since the

above approximation scheme only considers boundary conditions, the possible deviation error is not bounded.

The tile with seven trivariates, shown in Figure 10 (a), is paved and composed into the trivariate map shown in Figure 10 (b) to produce the micro-structure shown in Figure 10 (c). The tiles in this final micro-structure have degrees 12, 12 and 8 in the u, v and w directions, respectively. The region enclosed in the red rectangle in Figure 10 (c) is shown magnified in Figure 10 (d). The corresponding regions following an approximation using tri-quadratics and tri-cubics are shown in Figure 10 (e) and Figure 10 (f), respectively. G^1 -discontinuities in the surface are visible at the junction of two adjacent tiles in the approximations (see highlights). Yet, the quite-similar results, visually, can be explained by the fact that the input tiles are of low order typically and the deformation function is locally smooth.

7. Analysis using discontinuous tiles

Isogeometric Analysis is a paradigm introduced in [20] whose main feature is the adoption of shape functions used in computer aided design for also approximating physical fields and state variables involved in physical problems governed by partial differential equations (PDEs). Thus, geometry and approximate solution are both represented using spline functions. Trivariate spline tiles are especially useful when the models are to be subjected to isogeometric analysis [10].

In addition to the aim of improving the interoperability between design and analysis models, the properties of spline shape functions improve the performance of numerical analysis, with respect to finite element methods, for a wide variety of problems. For instance, spline functions easily provide high order approximation and highly simplify the refinement process. Additionally, their high continuity allows to tackle problems governed by PDEs involving high order derivatives and therefore requiring functions with high orders of continuity. Isogeometric analysis has been successfully applied to a large variety of physical problems, ranging from solid mechanics to fluid dynamics and electromagnetism (see [10] for further details and examples).

In this section, in order to show the robustness of the methodology presented in previous sections, heat transfer problems are studied in hierarchical micro-structures (recall Section 3) using tiles with discontinuities by means of isogeometric analysis techniques.

The steady state heat transfer problem is modeled by an elliptic PDE that describes the temperature distribution along the micro-structure, MS , under certain boundary conditions and heat sources/sinks. If the temperature distribution in MS is given by $s : MS \rightarrow \mathbb{R}$, then the Poisson equation that governs the problem reads:

$$-\nabla \cdot (k \nabla s) = f \quad \text{in } MS, \quad (9a)$$

$$s = g \quad \text{on } \Gamma_D, \quad (9b)$$

$$-k \nabla s \cdot \mathbf{n} = q \quad \text{on } \Gamma_N, \quad (9c)$$

$$-k \nabla s \cdot \mathbf{n} = h(s - s_0) \quad \text{on } \Gamma_R, \quad (9d)$$

where ∇ and $\nabla \cdot$ are the gradient and divergence operators, respectively; $k : MS \rightarrow \mathbb{R}$ is the material thermal conductivity coefficient; and $f : MS \rightarrow \mathbb{R}$ corresponds to the possible heat

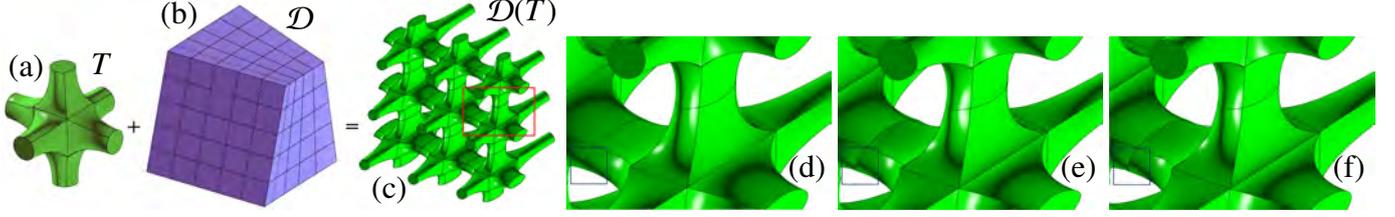


Figure 10: The tile with seven trivariates, T , shown in (a), is paved and composed into the trivariate map, \mathcal{D} , shown in (b) to produce the micro-structure, $\mathcal{D}(T)$, shown in (c). The degrees of T , \mathcal{D} and $\mathcal{D}(T)$ in (u, v, w) are $(3, 3, 2)$, $(1, 1, 2)$ and $(12, 12, 8)$, respectively. The region enclosed in the red rectangle in (c) is magnified in (d). The corresponding magnified micro-structures, approximated as tri-quadratics and tri-cubics, are shown in (e) and (f), respectively. Minute differences can be observed in highlights (Compare, for example, the blue rectangle zone in (d)-(f)) and the circular caps.

sources/sinks in the interior of the micro-structure. \mathbf{n} is the outwards normal vector on the boundary.

In addition, in the above equations, three different regions are considered in the boundary of the micro-structure, ∂MS : Γ_D is the Dirichlet boundary region, in this part of the boundary the temperature $g : \Gamma_D \rightarrow \mathbb{R}$ is a datum, i.e., it is strongly imposed in the analysis solution; the Neumann boundary region Γ_N , in which inlet/outlet heat fluxes $q : \Gamma_N \rightarrow \mathbb{R}$ are considered; and Γ_R , the Robin boundary region, at which the convective heat transfer between MS and the surrounding media, at a temperature s_0 , occurs, $h : \Gamma_R \rightarrow \mathbb{R}$ being the convection heat transfer coefficient. These three boundary regions are such as $\Gamma_N \cup \Gamma_D \cup \Gamma_R = \partial MS$ and $\Gamma_N \cap \Gamma_R = \Gamma_N \cap \Gamma_D = \Gamma_R \cap \Gamma_D = \emptyset$.

The PDE above is solved by means of a Galerkin method. Thus, in order to solve it, the approximate solution (the temperature distribution s), and test functions, are discretized by means of B-spline functions (see [10] for further details).

Recall that the hierarchical micro-structure MS is composed of a number of copies of the tile so that $MS = \bigcup_{i=0}^N \mathcal{D}(T_i)$, where \mathcal{D} is the deformation map and N is the number of copies of the tile. In the Galerkin method, the temperature distribution at every single tile $\mathcal{D}(T_i)$ is approximated using the B-spline shape function that describe the tile T_i (Equation (2)):

$$s_i(u, v, w) = \sum_{j_1=0}^{m_1} \sum_{j_2=0}^{m_2} \sum_{j_3=0}^{m_3} s_{i,j_1,j_2,j_3} B_{j_1}^{m_1}(u) B_{j_2}^{m_2}(v) B_{j_3}^{m_3}(w), \quad (10)$$

where s_{i,j_1,j_2,j_3} are the temperature coefficients associated to the basis functions for the i^{th} tile. In this way, every tile has a temperature distribution $s_i(u, v, w)$ that is discontinuous across tiles. The continuity of the temperature between adjacent tiles is enforced by setting up linear constraint equations between the temperature coefficients of the branch-out and branch-in faces of every level. For simplicity, we also assume the branch-out faces have no interior knots, or they are polynomials. Then, let

$$\mathcal{T}^k(u, v, w) = \sum_{i_1=0}^{n_1} \sum_{i_2=0}^{n_2} \sum_{i_3=0}^{n_3} P_{i_1,i_2,i_3}^k B_{i_1,\tau_1}^{n_1}(u) B_{i_2,\tau_2}^{n_2}(v) B_{i_3,\tau_3}^{n_3}(w), \quad (11)$$

be a tile at the k 'th level of the recursion and τ_i are the respective knot sequences. Let

$$\mathcal{T}^{k+1}(u, v, w) = \sum_{i_1=0}^{n_1} \sum_{i_2=0}^{n_2} \sum_{i_3=0}^{n_3} P_{i_1,i_2,i_3}^{k+1} B_{i_1,\tau_1}^{n_1}(u) B_{i_2,\tau_2}^{n_2}(v) B_{i_3,\tau_3}^{n_3}(w), \quad (12)$$

be a tile at the $k + 1$ 'th level of the recursion so that $\tau_1 = \mathbf{t}_1 \cup \{t_0^1, \dots, t_0^m\}$, where t_0^1, \dots, t_0^m are the multiplicity-order introduced knots in the u direction, in the different discontinuous

interior parameter locations, and same for the v direction. Because $\mathcal{T}^{k+1}(u, v, w)$ and $\mathcal{T}^k(u, v, w)$ share orders, and since all branch-out faces are polynomial patches by assumptions, the control points of every branch-out face can be expressed as a linear combination of the branch-in face, using the proper refinement matrix [7].

Thus, the temperature coefficients for all the tiles in the micro-structure are obtained by solving a single set of linear equations. This set of equations arises from the contributed equations of every tile $\mathcal{D}(T_i)$, plus the equations that enforce the continuity between the tiles.

Remark 2. By approximating $s_i, 1 \leq i \leq N$ in Equation (10) with the basis functions that describe the tile T_i (Equation (2)), instead of using the ones that describe $\bar{T}_i = \mathcal{D}(T_i)$ in Equation (3), the isoparametric paradigm is lost. However, this approach allows to keep a lower degree for the basis functions of the solution, which is critical for the analysis from the computational cost point of view: for the considered problem, the cost of the assembly using a standard procedure scales as $O(p^9)$, p being the degree of the basis functions [1].

Finally, it is important to remark that in order to preserve good approximation properties, the functional space used for the analysis must present, at every point, a continuity smaller than or equal, to that of the geometry parametrization of the micro-structure. This requirement is satisfied for the approach followed above in the construction of $\bar{T}_i = \mathcal{D}(T_i), 1 \leq i \leq N$.

8. Results

The framework described in Sections 3 to 7 is implemented in the IRIT [15] solid modeling kernel. We present additional results from our implementation in this section. Tiles comprising of surfaces, polygons and trivariates are used.

Figure 11 shows the construction of circular heat-sinks, following a heat-sink design for LEDs [3, 30]. The trivariate tiles, T_1, T_2 , in Figure 11, have (1,0) and (1,1) discontinuities respectively, along the (u, v) directions. The tile T_1 is composed into the circular twisted deformation map, \mathcal{D} , shown in Figure 11 (a), with repetition counts of (16, 1) along (u, v) directions and three levels of hierarchy along the w direction, resulting in the heat-sink shown in Figure 11 (b). Two similar constructions using tile T_2 with two levels of hierarchy are shown in Figure 11 (c) and (d). Two different values for scaling parameters along the w direction, as explained in Section 3.1, are used.

The next example demonstrates a sequence of two composition operations, applied one after another, creating two levels of (multi-resolution) micro-structures. The trivariate ele-

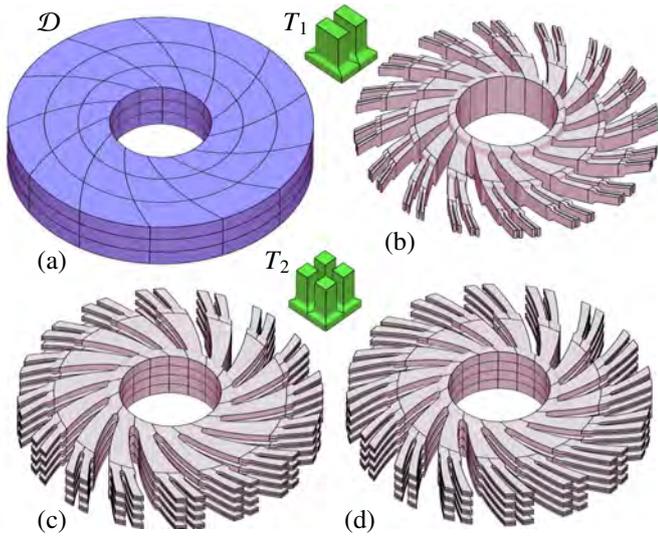


Figure 11: The tile T_1 is a trivariate with one discontinuity, along u direction. It is paved in the domain of the trivariate deformation map, \mathcal{D} , shown in (a) with repetition counts of $(16, 1)$ along (u, v) directions and three levels of hierarchy along the w direction. The resulting heat-sink, is shown in (b). A scaling of 0.5 along the w radial direction is used, as described in Section 3.1, resulting in the next level being half the height. The degrees of T_1 , \mathcal{D} and $\mathcal{D}(T_1)$ in (u, v, w) are $(2, 2, 3)$, $(3, 1, 2)$ and $(12, 12, 18)$, respectively. Two similar constructions using the tile, T_2 , having two levels of hierarchy are shown in (c) and (d), with w scaling value of 0.5 and 0.8, respectively. See also Figure 18.

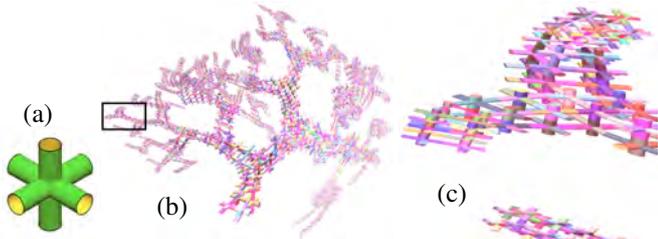


Figure 12: The trivariate elements of the hierarchical micro-structure shown in Figure 3 (d), with tri-quadratic approximation (see Section 6.3), are used as deformation maps for a bivariate tile with three open cylindrical surfaces is shown in (a). The tile in (a) is paved with repetition counts of $(2, 2, 2)$ along (u, v, w) to obtain the micro-structure shown in (b), with capping (see Section 6.2). The degrees of the surfaces in (u, v) , in the tile and in the final micro-structure, are $(3, 1)$ and $(18, 6)$, respectively. The region marked in the black rectangle in (b) is shown magnified in (c).

ments of the hierarchical micro-structure, constructed in Figure 3 (d), with tri-quadratic approximation (see Section 6.3), are employed again as deformation maps. This second level of micro-structures paves the bivariate tile from Figure 12 (a) $(2, 2, 2)$ times in each trivariate along (u, v, w) , only to be composed into the micro-structure shown in Figure 12 (b). Capping (see Section 6.2) is applied to all boundaries. The degrees of the surfaces in (u, v) , in the original tile and in the final micro-structure, are $(3, 1)$ and $(18, 6)$, respectively. The region marked in the black rectangle in Figure 12 (b) is shown magnified in Figure 12 (c).

Figure 13 shows more variants of micro-structures created with tiles with discontinuities. The tiles shown in Figure 13 (a) and (c) have one discontinuity along u and v . The tiles are paved with repetition counts of $(1, 1, 3)$ and composed into a deformation map to obtain the hierarchical structures shown in Figure 13 (b) and (d), respectively.

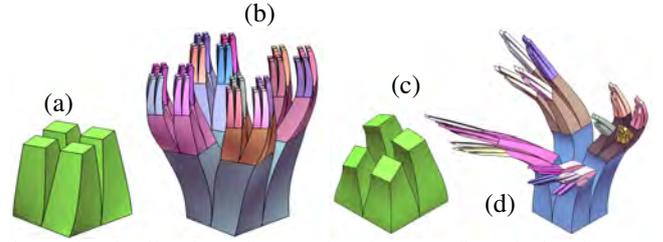


Figure 13: The tiles shown in (a) and (c) have one discontinuity along u and one along v . The tiles in (a), (c) are paved with repetition counts of $(1, 1, 3)$ and composed into deformation maps to obtain the hierarchical structures shown in (b) and (d), respectively.



Figure 14: Duck micro-structure of $(4, 4, 12)$ random tiles, each using an implicit tri-cubic trivariate B-spline with eight control points at each direction. A connectivity of two is ensured using the connectivity graph shown in Figure 5 (c). Note the vanishing tiles near the head and the tail.

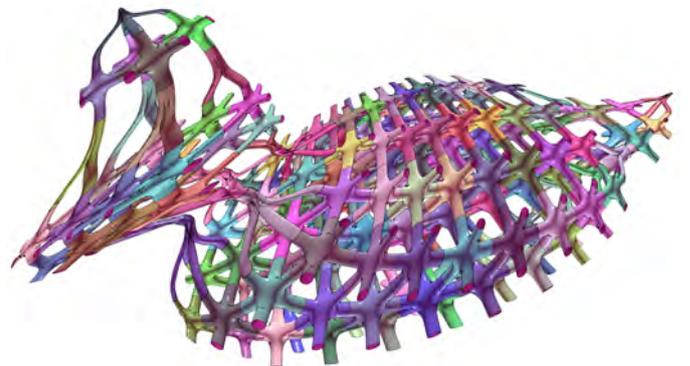


Figure 15: A Euclidean-uniform micro-structure with bifurcation and employing parametric tiles (from Figure 8) with $\epsilon = 0.35$. The length of the duck is little over 2.

Figure 14 shows micro-structures of $(4, 4, 12)$ random tiles composed into a duck trivariate. A tri-cubic trivariate B-spline with eight control coefficients at each direction is used for each implicit tile, and a connectivity two and C^1 continuity (recall Section 4.2.1) are ensured between the tiles.

The bifurcation tiles with parametric surfaces shown in Figure 8, are employed in Figure 15 to approximate uniform tiling in Euclidean space. Note the singular areas of the tip-of-the-head and the tail of the duck, that now have a single, relatively large, 1 to 4 tile each, covering those areas in the duck.

Figure 16 shows three examples of implicit micro-structures composed in a duck trivariate. In Figure 16 (a), an implicit tile of a 3D cross is uniformly tiled in the parametric space of the duck. Figure 16 (b) shows the same example as in (a) but with implicit bifurcation structures ($\epsilon = 0.35$, Algorithm 1),

approximating a uniform tiling in Euclidean space. Finally, Figure 16 (c) shows the same example as (b) but with randomization ($\alpha = 0.65$, Equation (6)). Note again the behavior near the head and tail.

A physical replica of random micro-structures composed in a duck trivariate is 3D printed from titanium and is shown in Figure 17, demonstrating the watertightness of the result.

Heat transfer analyses have been performed for two different heat-sink geometries using isogeometric analysis, as it was described in Section 7 (see Figure 18). The purpose of heat-sinks is to dissipate an inlet heat flux entering through the interior ring for the cylindrical sink and through the planar bottom face for the other one. All the other external faces dissipate the heat through the air by convection. There are no other internal heat sources/sinks.

In addition, considering that both heat-sinks can be fabricated using additive manufacturing, they can also be composed of a heterogeneous mix of aluminum and stainless steel. Thus, the resulting sink presents thermal properties that are a mix between both materials, whose conductivity coefficients are $205 \text{ W/m}^2 \text{ K}$ and $54 \text{ W/m}^2 \text{ K}$ for aluminum and stainless steel, respectively, where the units W, m and K stand for Watt, meter and Kelvin. The sink conductivity at every point, that is parameterized, as described in Section 6.1, is shown in Figures 18 (a) and (c) for both sinks: the surface in contact with the heat flux is pure aluminum (red color), whereas the percentage of stainless steel increases as it gets further from this surface (blue color).

Figures 18 (b) and (d) show the temperature distribution in both sinks also provided as a trivariate property over the same domain. As it can be appreciated, due to the convective heat dissipation, far from the inlet flux the sink temperature decreases towards the air temperature (30° C). The analyses have been performed using the isogeometric analysis library *igatools* [29].

All the examples described above were synthesized on a 3.4 GHz Intel i7 CPU with 32 GB RAM in a single thread mode and Windows 7 and took less than 5 seconds to generate. The memory footprints were less than 400 MB. The duck model with bifurcation tilings shown in Figures 5 and 16 took less than 4 seconds each to generate with a memory usage of 104 MB. The random duck micro-structure with guaranteed connectivity in Figure 14 took 108 seconds to construct with a memory usage 293 MB. The average number of iterations for the intra-cell connectivity algorithm (described in Section 4.2.2) is close to 20, where each iteration took, in average, less than 30 milliseconds. As for the inter-cell connectivity algorithm (described in Section 4.2.1), the number of iterations for ensuring the non-empty zero-set on each face on the connectivity graph is less than two, while each iteration took less than one millisecond, on average. The isogeometric analyses were run on a 3.0 GHz CPU with 32 GB of RAM in a single thread mode and Ubuntu 16.04. The computing time was 138 seconds for the circular heat-sink and 15 seconds for the vertical one, and the peak memory usages were 1651 MB and 334 MB, respectively.

9. Conclusion and future work

This paper proposes an extended set of primitives for the construction of hierarchical and random micro-structures, possibly with bifurcations. These primitives are based on Bézier/B-spline functional composition [11, 12] and extend the framework proposed in [14] by providing a richer interface, and enabling the creation of a wider spectrum of micro-structures. The

composition approach allows for a high degree of numerical precision, leading to water-tight models. The robustness of the structures thus created is verified by subjecting them to isogeometric analysis and to actual additive manufacturing.

We now identify some directions along which this work may be extended. While this work already provides preliminary support for implementing L-systems [31], there is more room for development in this direction leading to more elaborate hierarchical structures, including procedural modeling [26, 27]. Adding support for tiling where a tile may cross a knot line of the deformation map can potentially widen the scope of application of this work. While performing degree reduction, better continuity and smoothness of the surfaces and trivariates can improve the quality of the results. In this work we discussed fixed shaped tiles or randomized tiles but one can envision other tiling schemes, like general regular 2D and 3D tiling [34] etc. The connectivity graph used to generate connected implicit micro-structures is constructed using up to two disjoint spanning trees L_1 and L_2 . However, finding L_2 is not always guaranteed, and also depends on L_1 . More efficient solutions for finding the maximal number of spanning trees in a grid graph might be considered [5, 24]. While the connectivity graph construction itself is not a main focus in this paper, other connectivity graphs that reflect other desired properties (such as orientation prescriptions) on the connectivity of the generated micro-structures might be studied as well. Finally, the synthesis of each tile is independent, and hence one may trivially parallelize the composition of tiles into the deformation map, achieving significant speedups.

Acknowledgments

This research was supported in part with funding from the Defense Advanced Research Projects Agency (DARPA), under contract HR0011-17-2-0028. The views, opinions and/or findings expressed are those of the author and should not be interpreted as representing the official views or policies of the Department of Defense or the U.S. Government.

We thank the reviewers for their comments and suggestions that helped to improve this work.

References

References

- [1] Antolin, P., Buffa, A., Calabrò, F., Martinelli, M., Sangalli, G., 2015. Efficient matrix computation for tensor-product isogeometric analysis: The use of sum factorization. *Computer Methods in Applied Mechanics and Engineering* 285 (Supplement C), 817 – 828.
- [2] Armillotta, A., Pelzer, R., Nov 2008. Modeling of porous structures for rapid prototyping of tissue engineering scaffolds. *The International Journal of Advanced Manufacturing Technology* 39 (5), 501–511.
- [3] Bornoff, R., Parry, J., March 2015. An additive design heatsink geometry topology identification and optimisation algorithm. In: 2015 31st Thermal Measurement, Modeling Management Symposium (SEMI-THERM). pp. 303–308.
- [4] Burczyński, T., Kuś, W., 2009. *Microstructure optimization and identification in multi-scale modelling*. Springer Netherlands, Dordrecht, pp. 169–181.
- [5] Chan, W.-T., Chin, F. Y. L., Ting, H.-F., 2000. Escaping a grid by edge-disjoint paths. In: *Proceedings of the Eleventh Annual ACM-SIAM Symposium on Discrete Algorithms. SODA '00*. Society for Industrial and Applied Mathematics, Philadelphia, PA, USA, pp. 726–734.

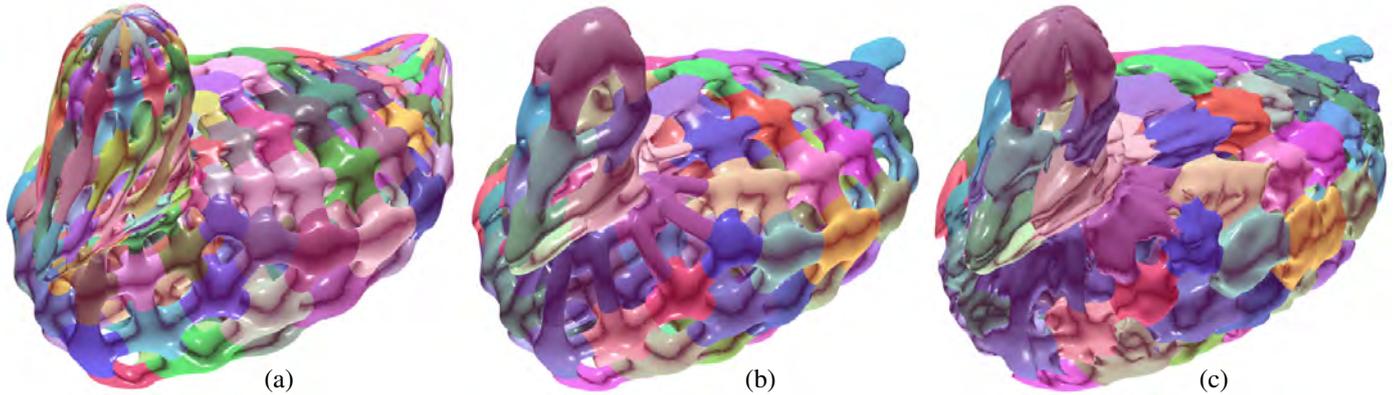


Figure 16: Implicit micro-structures composed in a duck trivariate. (a) presents an implicit uniform tiling in the parameteric space. (b) presents an implicit uniform tiling in Euclidean space with implicit bifurcation structures ($\epsilon = 0.35$, and the length of the duck is little over 2). (c) is similar to (b) but with randomization ($\alpha = 0.65$). Compare (c) with Figure 14.



Figure 17: A physical 3D printed random duck. See also Figure 14.

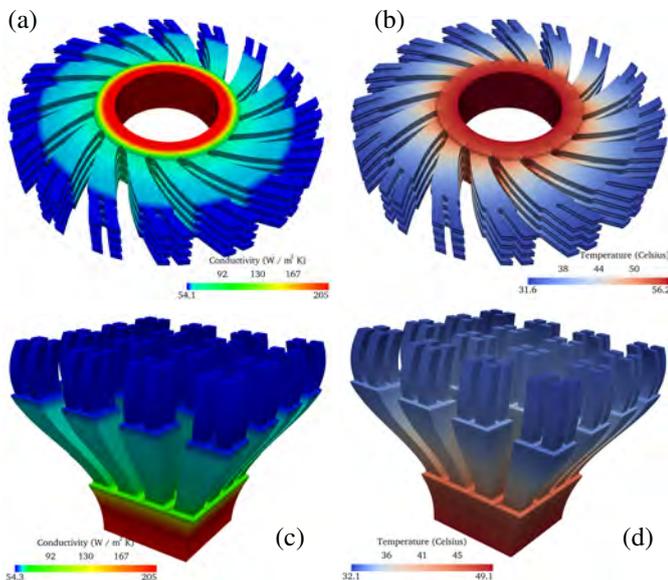


Figure 18: Heat transfer analysis in two different heterogeneous heat-sinks, using isogeometric analysis. In (a) and (c), the thermal conductivity k at every point is shown, color-coded. This heterogeneous coefficient is defined at every point according to the description given in Section 6.1. (b) and (d) show the temperature distribution resulting from the analyses performed in both heat-sinks. Both cases present similar boundary conditions: an inlet heat flux ($q = 1515.8 \text{ W/m}^2$) in the interior part of the ring for (b) and in the bottom planar face of (d); and convective heat transfer between the sink and the surrounding air through all the other external faces (air temperature $u_0 = 30^\circ \text{ C}$ and convection coefficient $h = 5 \text{ W/m}^2 \text{ K}$). See also Figure 11.

- [6] Chu, C., Graf, G., Rosen, D. W., 2008. Design for additive manufacturing of cellular structures. *Computer-Aided Design and Applications* 5 (5), 686–696.
- [7] Cohen, E., Riesenfeld, R. F., Elber, G., 2001. Geometric modeling with splines, an introduction. A K Peters.
- [8] Conde-Rodríguez, F., Torres, J.-C., García-Fernández, Á.-L., Feito-Higueruela, F.-R., Jan 2017. A comprehensive framework for modeling heterogeneous objects. *The Visual Computer* 33 (1), 17–31.
- [9] Cormen, T. H., 2009. Introduction to algorithms. MIT press.
- [10] Cottrell, J. A., Hughes, T. J. R., Bazilevs, Y., 2009. Isogeometric analysis: Toward integration of CAD and FEA. Wiley.
- [11] DeRose, T. D., Goldman, R. N., Hagen, H., Mann, S., Apr. 1993. Functional composition algorithms via blossoming. *ACM Trans. Graph.* 12 (2), 113–135.
- [12] Elber, G., 1992. Free form surface analysis using a hybrid of symbolic and numerical computation. PhD Thesis, University of Utah.
- [13] Elber, G., 2005. Geometric texture modeling. *IEEE Computer Graphics and Applications* 25 (4), 66–76.
- [14] Elber, G., 2016. Precise construction of micro-structures and porous geometry via functional composition. In: *Proceedings of the 9th International Conference on Mathematical Methods for Curves and Surfaces*. pp. 108–125.
- [15] Elber, G., January 2017. Irit modeling environment. "<http://www.cs.technion.ac.il/~irit/>".
- [16] Elber, G., Kim, Y.-J., Kim, M.-S., 2012. Volumetric boolean sum. *Computer Aided Geometric Design* 29 (7), 532 – 540, *Geometric Modeling and Processing* 2012.
- [17] Farin, G., 1993. Curves and surfaces for computer aided geometric design. Academic Press Professional, Boston.
- [18] Farouki, R. T., Rajan, V. T., Jun. 1988. Algorithms for polynomials in Bernstein form. *Comput. Aided Geom. Des.* 5 (1), 1–26.
- [19] Gao, W., Zhang, Y., Ramanujan, D., Ramani, K., Chen, Y., Williams, C. B., Wang, C. C., Shin, Y. C., Zhang, S., Zavattieri, P. D., 2015. The status, challenges, and future of additive manufacturing in engineering. *Computer-Aided Design* 69 (Supplement C), 65 – 89.
- [20] Hughes, T. J. R., Cottrell, J. A., Bazilevs, Y., 2005. Isogeometric analysis: CAD, finite elements, NURBS, exact geometry and mesh refinement. *Computer-Aided Design* 194, 4135–4195.
- [21] Kendall, W. S., van Lieshout, M., 1998. Stochastic geometry: Likelihood and computation. Chapman and Hall/CRC, London.
- [22] Leblanc, L., Houle, J., Poulin, P., Apr 2011. Modeling with blocks. *The Visual Computer* 27 (6), 555. URL <https://doi.org/10.1007/s00371-011-0589-4>
- [23] Lorensen, W. E., Cline, H. E., Aug. 1987. Marching cubes: A high resolution 3d surface construction algorithm. *SIGGRAPH Comput. Graph.* 21 (4), 163–169.
- [24] Marx, D., 2004. Eulerian disjoint paths problem in grid graphs is np-complete. *Discrete Applied Mathematics* 143 (1), 336 – 341.
- [25] Medeiros e Sá, A., Mello, V. M., Rodriguez Echavarría, K., Covill, D., Jun 2015. Adaptive voids. *The Visual Computer* 31 (6), 799–808.

- [26] Müller, P., Wonka, P., Haegler, S., Ulmer, A., Van Gool, L., Jul. 2006. Procedural modeling of buildings. *ACM Trans. Graph.* 25 (3), 614–623. URL <http://doi.acm.org/10.1145/1141911.1141931>
- [27] Parish, Y. I. H., Müller, P., 2001. Procedural modeling of cities. In: *Proceedings of the 28th Annual Conference on Computer Graphics and Interactive Techniques. SIGGRAPH '01.* ACM, New York, NY, USA, pp. 301–308. URL <http://doi.acm.org/10.1145/383259.383292>
- [28] Pasko, A., Fryazinov, O., Vilbrandt, T., Fayolle, P.-A., Adzhiev, V., 2011. Procedural function-based modelling of volumetric microstructures. *Graphical Models* 73 (5), 165 – 181.
- [29] Pauletti, M. S., Martinell, M., Cavallini, N., Antolin, P., 2015. Igatools: An isogeometric analysis library. *SIAM Journal on Scientific Computing* 37 (4), C465–C496.
- [30] RapidLED, 2017. Single LED heat sinks. "<https://www.rapidled.com/single-led-heat-sink/>".
- [31] Rozenberg, G., Salomaa, A., 1992. *Lindenmayer Systems: Impacts on Theoretical Computer Science, Computer Graphics, and Developmental Biology.* Springer.
- [32] Schroeder, C., Regli, W. C., Shokoufandeh, A., Sun, W., 2005. Computer-aided design of porous artifacts. *Computer-Aided Design* 37 (3), 339 – 353, heterogeneous Object Models and their Applications.
- [33] Wang, H., Chen, Y., Rosen, D., January 2005. A hybrid geometric modeling method for large scale conformal cellular structures. In: *Volume 3: 25th Computers and Information in Engineering Conference, Parts A and B.*
- [34] Wikipedia, January 2018. Euclidean tilings by convex regular polygons.
- [35] Xiao, F., Yin, X., 2016. Geometry models of porous media based on voronoi tessellations and their porosity-permeability relations. *Computers and Mathematics with Applications* 72 (2), 328 – 348, the Proceedings of ICMMS 2014.