Research article

IJAC

A generalized framework for designing topological interlocking configurations

International Journal of Architectural Computing
2019, Vol. 17(1) 53–73

DOI: 10.1177/1478077119827187
journals.sagepub.com/home/jac
SAGE

Andres Bejarano
and Christoph Hoffmann

Abstract

A topological interlocking configuration is an arrangement of pieces shaped in such a way that the motion of any piece is blocked by its neighbors. A variety of interlocking configurations have been proposed for convex pieces that are arranged in a planar space. Published algorithms for creating a topological interlocking configuration start from a tessellation of the plane (e.g. squares colored as a checkerboard). For each square S of one color, a plane P through each edge E is considered, tilted by a given angle θ against the tessellated plane. This induces a face F supported by P and limited by other such planes nearby. Note that E is interior to the face. By adjacency, the squares of the other color have similarly delimiting faces. This algorithm generates a topological interlocking configuration of tetrahedra or antiprisms. When checked for correctness (i.e. for no overlap), it rests on the tessellation to be of squares. If the tessellation consists of rectangles, then the algorithm fails. If the tessellation is irregular, then the tilting angle is not uniform for each edge and must be determined, in the worst case, by trial and error. In this article, we propose a method for generating topological interlocking configurations in one single iteration over the tessellation or mesh using a height value and a center point type for each tile as parameters. The required angles are a function of the given height and selected center; therefore, angle choices are not required as an initial input. The configurations generated using our method are compared against the configurations generated using the angle-choice approach. The results show that the proposed method maintains the alignment of the pieces and preserves the co-planarity of the equatorial sections of the pieces. Furthermore, the proposed method opens a path of geometric analysis for topological interlocking configurations based on non-planar tessellations.

Keywords

Topological interlocking, surface tessellation, irregular geometry, parametric design, convex assembly

Introduction

A topological interlocking configuration (TIC) is an assembly of building blocks with interfaces relying on shape and alignment. Such blocks are connected solely by face contact, disregarding the use of adhesives, connectors, or any other binding mechanism for keeping pieces together. The structure holds itself by the

Department of Computer Science, Purdue University, West Lafayette, IN, USA

Corresponding author:
Andres Bejarano, Department of Computer Science, Purdue University, 305 N University Street, West Lafayette, IN 47907, USA.
Email: andresbeja87@outlook.com

kinematic constraints imposed by both a peripheral support structure and the neighborhood surrounding every piece in the configuration. The principles behind TICs were discovered during the Renaissance in a time when both designs and construction techniques for floors and flat vaults were a signal of architectural innovation. The name was coined in the early 2000s when the material properties of TICs as well as their applications started to be formally approached.

The kinematic constraints on TICs imply two different considerations while designing a configuration: each piece individually and the entire configuration. At the individual level, each piece prevents motion on the pieces in its vicinity. That is, the shape of an individual piece must be designed in such a way that both push and pull motions of the piece are blocked by the wedging actions of the neighboring pieces. However, when the configuration is taken as a single entity, it is required to place a peripheral support frame that guarantees the configuration does not collapse. Such frame is required for both planar and curvilinear surfaces in order to prevent lateral motions and provide support for the pieces at the boundary of the configuration.

Generating a TIC starts with the tessellation of a surface. For every tile in the tessellation, there is a correspondent piece generated such that its shape prevents push and pull movements for itself and its neighboring pieces. A correct set of parameters must be either given or be found to generate a TIC with no piece overlapping. In the classic generation framework, it is required to set a rotation angle for each edge in the tessellation. Unless the tessellation is a checkerboard, the problem increases in complexity as the tessellation becomes less regular. Due to such generalization, not all angle setups work properly. The algorithms proposed for solving such problems require multiple iterations over the tiles in the tessellation, starting with an initial estimation of angles, which are improved and checked on each iteration. As usual with such types of algorithms, a stop condition must be defined followed by a check of correctness for the resultant configuration.

Our proposed TIC generation framework is aimed toward two major goals: first, finding such rotation attributes on a single iteration, and second, extending the algorithm for generating TICs based on three-dimensional (3D) domains such as solids, meshes, and parametric surfaces. Rather than starting with a set of estimated angles, we propose to define a single distance parameter and a selected face center for finding an entire valid angle setup in a single iteration. The resultant framework is then a generalization of the classic checkerboard method and its variants for different surface tessellations.

Previous work

The fundamental idea behind TICs can be traced back to the late 17th century when flat vault designs were a sign of innovation. In 1699, Joseph Abeille introduced an ashlar-type block with the shape of an isosceles trapezium which, when repeated in series, realizes a flat vault that supports itself when the assembly process is finished.[1] Such approach required a single block shape, and it was a novel approach in a time when vault designs relied on different shapes. Improvements and later developments based on Abeille's design included the removal of voids and the use of different construction materials. An extended documentation about Abeille's bond along with its historical and architectural relevance can be found in the works of Fallacara,[1] Brocato and Mondardini,[2,3] and Brocato et al.[4]

Nearly 300 years later, the same principle was considered for the design of vertically interlocked paving. In 1984, Michael Glickman argued that traditional interlocking paving blocks were large, heavy, prone to damage due to connections, and difficult to install. He proposed a block that is easy to both manufacture (no joinery) and transport, and unlikely of accidental damage while handling.[5] The resulting block was a truncated tetrahedron posed over one of the tetrahedron edges, with its equatorial section must be at equal distance from both top and bottom faces obtained after truncation. Additional spacer ribs were embedded to the block for load equalization.

Early in the 21st century, a third independent approach considered the very same principle applied to the design of materials and structures. In 2001, Dyskin et al.[6] considered the concept of key block while

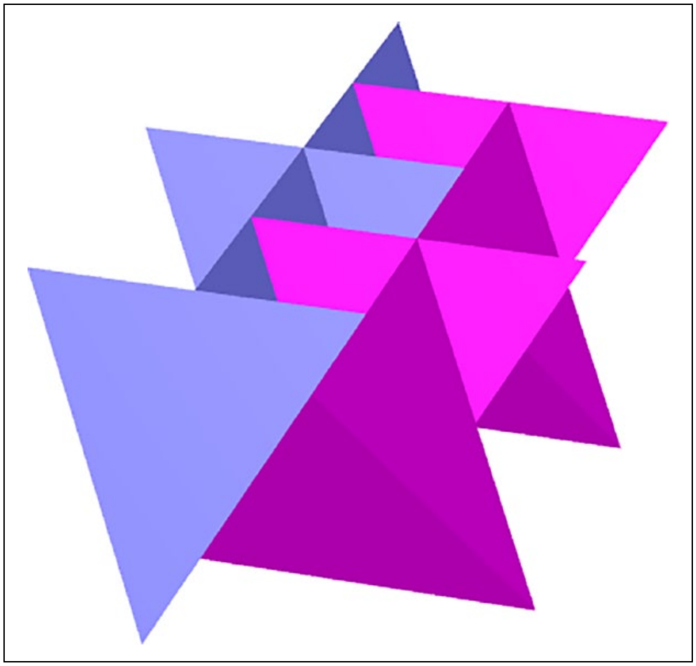


Figure 1. A general view of self-interlocking tetrahedra. Two column tetrahedra sections are displayed with different colors.

explaining the kinematic constraints among pieces and how blocks are held in place. The authors concluded that an assembly of identical tetrahedra, as shown in Figure 1, contains self-interlocking; that is, the key blocks for one section (i.e. column tetrahedra) are blocked by parallel cross sections where they are no longer key blocks. Such type of configurations requires peripheral constraint structures to prevent lateral strains. Later that same year, Dyskin et al.[7,8] approached the problem of crack propagation and how the topology of subdivided materials can solve it. The authors expanded the principle of self-interlocking tetrahedra using both octagon-based and circle-based shapes for the construction of planar sections.

Geometrical analysis of TICs had a significant role as new physical properties were discovered. In 2003, Dyskin et al.[9] reported that the Platonic solids can maintain topological interlocking when assembled in planar sections. A tetrahedron is obtained from the evolution of a planar squared tile toward both halfway up and down until a grid is formed. Such grid is the dual of the initial planar squared tiling of the surface. Using a regular hexagonal tiling on a planar surface allows the generation of TICs based on hexahedra, octahedra, and dodecahedra depending on the orientation of the equatorial sections of the solids (Figure 2 shows the required orientation for such solids). A TIC based on icosahedra is possible by starting with the tessellation composed of decagons and darts, where the latter shape is considered as an empty space which does not generate an interlocking piece. Moreover, authors give an example of a TIC based on truncated icosahedra, such example opens the geometrical analysis on TICs using more complex convex shapes such as the Archimedean solids and the Catalan solids. The basic premise for convex interlocking is that an alignment of certain convex polyhedra can be arranged in such a way that the resultant configuration is self-interlocking. The formalization of the piece generation process starting from a regular planar tessellation was introduced by Kanel-Belov et al.,[10] and the development of such process is presented on section “TIC generation frameworks.”

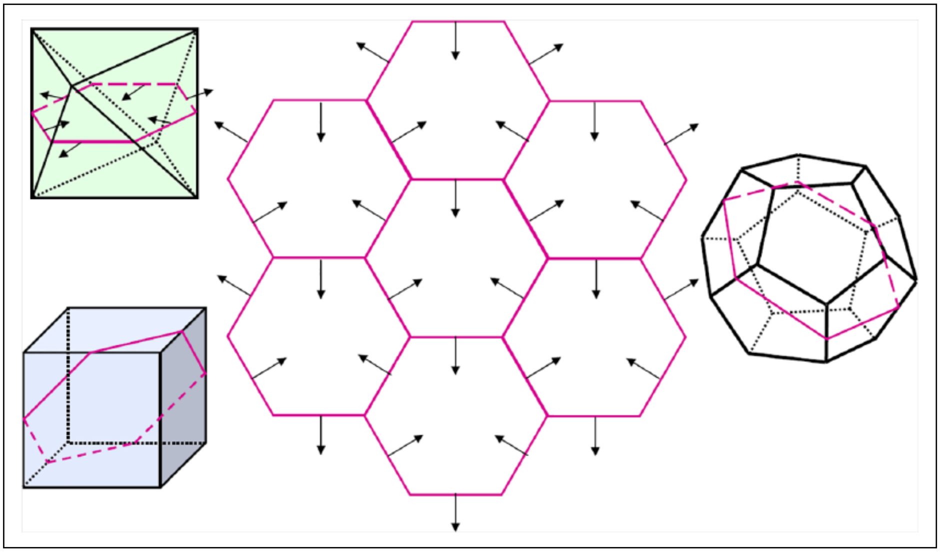


Figure 2. Hexagonal tiling with associated Platonic solids.
Source: Kanel-Belov et al.[10]

The generation of the TICs based on non-planar surfaces has been in scope for several researchers in the area. Regarding the already mentioned Abeille's bond, Brocato and Mondardini[2,3] considered the analysis and generation of spherical vaults using such bond. Their approach focused on the optimal parameters for the stereotomy of the pieces as well as a finite model analysis for mechanical purposes. The resultant shapes are nexorades, a "structure made of nexors, a beam often having four simple connections, two at its ends to be supported and two at intermediate points to bear other nexors."[2] Classical nexors are truncated irregular tetrahedra. When the nexors are built from regular tetrahedra, it is notorious their resemblance with the proposed designs previously reported by Glickman and Dyskin.

The design of TICs based on non-planar surfaces has been of interest of researchers in the topic. Tessmann[11] reported the result of student projects who built TICs as geometrical differentiated, reversible, force-locked systems. The solutions presented by the students considered a variety of shapes clearly derived from tetrahedra; although not convex in nature, the faces were kept planar for preserving the fundamental notion behind the topological interlocking principle. The design of a valid border for a finite configuration was also considered, offering an alternative solution from an architectural perspective. Such solution involved windmill shapes for constraining the boundary, allowing the configuration to end on an edge without additional peripheral structure. For curvilinear surfaces, Weizmann et al.[12] adapted the generation method initially formulated by Kanel-Belov et al.[10] A planar tessellation is projected onto the curvilinear surface and then the geometry of the pieces is built in accordance to the curvature of the base surface.

Recent advances concerning geometrical approaches are related to computational architecture. Weizmann et al.[13] used TICs for building facades, extending the catalog of resultant pieces based on semi-regular and non-regular tessellations (examples of TICs from semi-regular tessellations in Figure 3). Their approach is a direct application of the method described by Kanel-Belov et al. For tessellations inscribed on curvilinear surfaces resulted on valid configurations even when curvature is presented, the same idea is applied for the design and construction of floors alongside the design of additional structures required for the assembly process.[12] The results were subject to structural simulation for load-carrying capacity and deflection analysis. Finally, Weizmann et al.[14] introduced a computational method for two-dimensional (2D) patterns generation used for TIC generation.

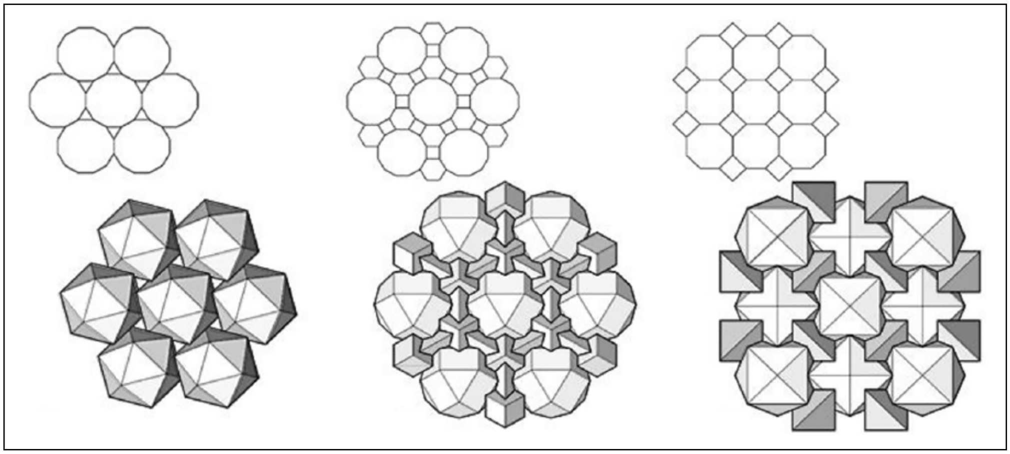


Figure 3. Semi-regular tessellations and respective TICs.
Source: Weizmann et al.[13]

Finally, there are researchers looking for general interlocking frameworks suitable for representing configurations based on different binding mechanisms. Wang et al.[15] proposed a generalized framework that represents interlocking assemblies based on different types of joinery (e.g. voxelized pieces, plates, and frame structures). The framework assumes each element in the assembly is rigid, the neighboring parts have planar contacts only, elements are removed by translational motions only, and all other elements remain in place after removing a part. The framework guarantees that all parts and part groups are locked for all directions except for a key piece. The relationship between elements is represented using *Directional Block Graphs*; such structure helps to avoid the complexity of searching operations through the assembly. Furthermore, the authors propose an algorithm for constructing the geometry of the pieces using the appropriate type of binding joints.

TIC generation frameworks

Building a TIC requires the generation of a block for each individual tile or polygon in the surface tessellation. As described in Kanel-Belov et al.,[10] a quadrilateral generates a tetrahedron; a hexagon generates a cube, octahedron, or a dodecahedron; a decagon generates a dodecahedron or an icosahedron. For polygons with higher number of sides, it is possible to obtain antiprisms as interlocking pieces, examples of these can be found in Weizmann et al.[12–14]

We approach the problem of TIC generation from a computational perspective. In this section, we first describe an algorithmic formulation for the geometric representations of the problem. Then, we discuss the overall TIC generation method used by the authors previously mentioned, we focus on an algorithmic perspective. Finally, we present a method for TIC generation that requires only a single iteration over the input.

Algorithmic representation

The generation of a TIC begins with a surface tessellation composed of convex tiles. The tessellation is equivalent to a mesh, essentially a graph, for computational purposes. A mesh is then represented using a doubly connected edge list (DCEL), a classic data structure considered for geometric purposes.[16] The geometric elements are stored in three main lists: vertices, faces, and halfedges. The incidence references are

Table 1. Algorithmic notation of the attributes of the DCEL elements stored in the (a) vertices, (b) faces, and (c) halfedges lists.

vertices[v].coordinates vertices[v].halfedge	faces[f].halfedge	halfedges[h].start halfedges[h].end halfedges[h].previous halfedges[h].next halfedges[h].twin halfedges[h].face
(a)	(b)	(c)

DCEL: doubly connected edge list.

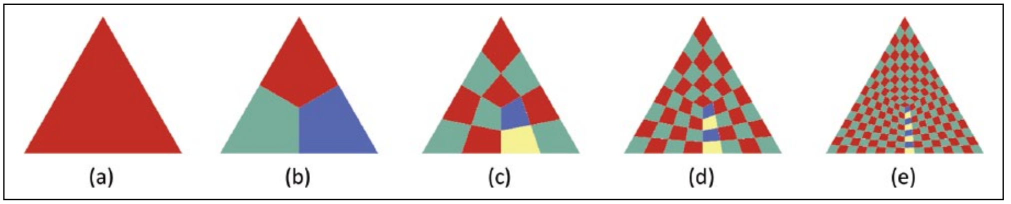


Figure 4. Midpoint subdivisions of a triangle: (a) initial, (b) one subdivision, (c) two subdivisions, (d) three subdivisions, and (e) four subdivisions.

accessed as attributes of the elements. Vertices, faces, and halfedges elements are indexed in their respective list. Table 1 shows the algorithmic representation of the element attributes stored in the described lists.

Face subdivision

The TIC generation methods to be described work on meshes whose faces have an even number of edges. An effective procedure to pre-process faces that have an odd number of faces is by subdividing the face into quadrilaterals using the midpoint subdivision method:[17] let $\mathrm{DCEL}(M) = \{\mathrm{vertices}, \mathrm{faces}, \mathrm{halfedges}\}$ be the DCEL representation of the mesh $M = \{V, F\}$ containing the information about the geometric domain. Let $f \in$ faces be a face of the mesh; let n_f be the number of sides of the face; and let C_f be a center point within the face, it could be the barycenter, centroid, or any other center point that can be calculated for all faces (e.g. the diagonals intersection on quadrilateral faces). For every edge $e \in f$, get its midpoint e_m and define the line segment between e_m and C_f, such process subdivides the face into n_f quadrilaterals. Explicit bookkeeping indicating the new faces and the incidence between the halfedges is required when using a DCEL for mesh representation. Figure 4 shows an example of consecutive midpoint subdivisions starting with an equilateral triangle.

Tilting angle method

We name tilting angle method (TAM) to the TIC generation approach described by Kanel-Belov et al.[10] The method works on even-sided polygonal faces with alternating direction values on the edges of the faces. Through every edge, there is a plane P, initially orthogonal to the tessellation plane. P is tilted out of the perpendicular by a tilting angle, in the indicated direction value. The intersection of the tilted planes through the edges of each tessellation tile defines the edges and vertices of a polyhedron.

The steps of the TAM are explained using a checkerboard tessellation as an example. An arrow (indicating the tilting direction and value) is assigned to each edge according to the color of the tile: a dark tile has both north and south arrows pointing outward the tile while both east and west arrows point inward the tile;

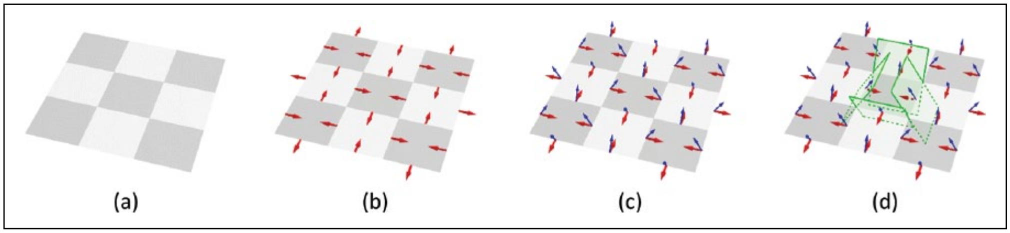


Figure 5. Steps of the TAM on a checkerboard: (a) initial checkerboard, (b) arrow setup (red), (c) tilted direction arrows (blue), and (d) incident tilted planes for the middle tile (green).

similarly, a light tile has both north and south arrows pointing inward the tile while both east and west arrows point outward the tile. All arrows must comply with the following rules: they have the same length, they start on the midpoint of the respective edge, and they are perpendicular to the edge in the tessellation plane. The arrows indicate the chosen tilting angle by their length and the tilting direction by their direction (Figure 5).

A generalized version of the tilting method can be devised for plane tessellations composed of even-sided tiles.[10] The direction arrows are assigned to the edges such that the inward and outward directions alternate on the edges of every tile. On each edge, a tiling angle value is set, and a tilted plane as described for the checkerboard case. The intersection of the tilted planes incident to a tile generates a polyhedron as before. Figure 2 shows examples of the arrow setup applied to a hexagonal tessellation and the piece alignment with respect to the tiles for representing the TIC using cubes, octahedra, or dodecahedra.

We describe the TAM using an algorithmic formulation. Let $\mathrm{DCEL}(M) = \{\mathrm{vertices}, \mathrm{faces}, \mathrm{halfedges}\}$ be the DCEL representation of the mesh $M = \{V, F\}$ containing the information of the geometric domain. Let $d \in \{-1,1\}$ be the initial direction value to be used. Traverse through each face $f \in \mathrm{faces}$ and set the alternating direction values on the halfedges of the respective face, such directions are set as $\mathrm{halfedges}[h].\mathrm{direction} = d$ and $\mathrm{halfedges}[h].\mathrm{twin.direction} = -d$. Alternate d and $-d$ each time a direction value is set on a halfedge of the face. For every halfedge, define $\mathrm{halfedges}[h].\mathrm{midpoint} = h_m$ and $\mathrm{halfedges}[h].\mathrm{twin.midpoint} = \mathrm{halfedges}[h].\mathrm{midpoint}$, where h_m is the midpoint of the edge. Let $C_f = \mathrm{center}(\mathrm{faces}[f])$ be a center point of the current face. The direction arrow on each edge is described by the unit vector

$$\mathrm{halfedges}[h].\mathrm{vector} = \frac{PC_f}{\|PC_f\|}$$

where P is a point along $\mathrm{halfedges}[h]$ such that PC_f and the direction vector of the halfedge are orthogonal. Since the direction vector is associated with an edge, it is valid to set $\mathrm{halfedges}[h].\mathrm{twin.vector} = \mathrm{halfedges}[h].\mathrm{vector}$. The vectors associated with each halfedge are rotated with respect to the direction vector of the respective halfedge. Then

$$\begin{aligned}&\mathrm{halfedges}[h].\mathrm{tilted}\\ &= \mathrm{rotate}\left(\mathrm{halfedges}[h].\mathrm{vector}, \mathrm{direction}\left(\mathrm{halfedges}[h]\right), \alpha\ \mathrm{halfedges}[h].\mathrm{direction}\right)\end{aligned}$$

where

$$\mathrm{rotate}(V, K, \theta) = V\cos\theta + (K \times V)\sin\theta + K(K \cdot V)(1-\cos\theta)$$

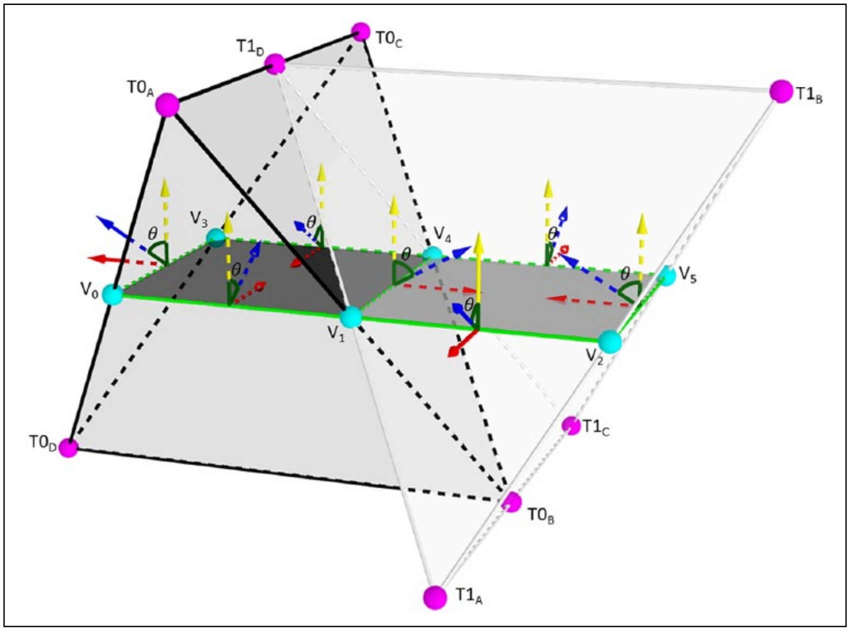


Figure 6. Elements used by TAM for generating the pieces. Cyan points are the vertices of two squares in the tessellation. Red arrows describe the directions associated with the edges of the squares. Yellow arrows are the normal vectors of the edges. Blue arrows are the rotated vectors that describe the tilted planes by an angle θ. Magenta points are the vertices of the resultant interlocking pieces T0 and T1 from left and right squares, respectively.

is the vector V' representing vector V rotated with respect to the unit axis vector K by the angle θ. This expression is known as the axis–angle rotation or Rodrigues' rotation formula. Let $\text{plane}(\text{halfedges}[h]) = \{a, b, c, d\}$ be the components of the plane incident to the halfedge $\text{halfedges}[h]$. The vertices of the resultant interlocking block are defined by the intersection of planes associated with three consecutive halfedges of a face. Then, a vertex

$$v = \text{intersect}(\text{plane}(\text{halfedges}[h].\text{previous}), \text{plane}(\text{halfedges}[h]), \text{plane}(\text{halfedges}[h].\text{next}))$$

where $\text{intersect}(A, B, C)$ is the intersection point of the planes A, B, and C assuming the ranks of the coefficient matrix and the augmented matrix from the linear system defined by the components of the three planes is equal to 3, which means the three planes intersect at a point. Figure 6 shows the elements, from two neighboring squares, used by TAM for generating the respective interlocking pieces.

A TIC is valid when the constructed polyhedra keep the interlocking behavior and do not overlap with each other. The resultant TICs generated by the TAM are valid under certain conditions. First, the faces of the tessellation mesh must be regular, and second, the faces must exist in the same plane. The interlocking behavior guaranteed by the TAM is formally established in Kanel-Belov et al.[10] When the faces of the mesh are not regular, or the faces are not coplanar to each other, then overlapping artifacts appear in the resultant configuration. Building a TIC using the TAM on meshes with non-regular faces will generate an invalid configuration due to overlapping polyhedra. In the original description of the method, the same tilting angle α is used for determining the incident plane for each edge of the faces. But using the same angle value for all edges does not guarantee that the resultant configuration will be valid. An example of such situation is a TIC based on a rectangular mesh and generated using the TAM; Figure 7 shows the resultant TIC using $\alpha = \pi / 3$ on the edges. The resultant TIC shows two issues:

1. Overlapping polyhedra (the resultant tetrahedra overlap along a row or a column of the rectangular mesh);
2. Polyhedral misalignment (resultant tetrahedra are not aligned properly such that their equatorial sections are not incident to their respective tiles).

Both issues are solved by forcing the equatorial section of a piece to be incident to the respective face in the mesh. Referring to Figure 8, let $f \in F$ be a face of the mesh, let C_f be the chosen center point within the face, and let N_f be the normal of the face. Assume that the vertices of the face are coplanar, that is, the face exists in a plane. Let T_f and B_f be the respective points on the top and bottom sections of the resultant piece found along the line defined by C_f and N_f, that is, $T_f = C_f + \lambda N_f$ and $B_f = C_f - \lambda N_f$ for some scalar $\lambda \in \mathbb{R}, \lambda > 0$, respectively. The resultant piece from *f* is valid for a TIC if its equatorial section exists on the same plane as *f*, which is equivalent to $\mathrm{dist}(C_f, T_f) = \mathrm{dist}(C_f, B_f)$, where $\mathrm{dist}(A, B)$ is the Euclidean distance between points *A* and *B*. Then, the resultant piece is valid if its both top and bottom sections are at the same distance with respect to C_f and both T_f and B_f are found in their respective sections. For a tetrahedron, such sections are the top and bottom edges, for an antiprism, such sections are the top and bottom faces which are rotated a π/n angle with respect to each other, where *n* is the number of sides of the even-sided polygonal tile.

The previous observation leads to the definition of a required constraint for interlocking tetrahedra and antiprisms: the vertices of a piece must lie on their respective top or bottom sections of the piece. Since each vertex is calculated as the intersection of the tilted incident planes from three consecutive edges in a face, the constraint must be represented by a parameter during the definition of such planes. For the TAM, each plane is defined by the rotated direction vector and the vertices of an edge. Since the vertices of the face are given, the rotation angle is the remaining parameter left for implementing the required constraint. It means that for every edge *e* in a face, the rotation angle α_e must be calculated such that the vertex determined by the intersection of three consecutive tilted planes comply with the constraint. For tessellations using squares, it is known that the same angle value applies to all edges; however, such statement does not hold for other quadrilaterals or even-sided polygons. The definition of such angles could be left as both user input or the output of an algorithmic search for valid tilting angles on each edge. Figure 8 shows the geometric nature of the problem for the case of a rectangular face.

The TAM also generates non-valid TICs when the incident faces on an edge are not coplanar. Let $a, b \in F$ be two faces in a mesh both incident to a common edge *e*, and let θ_e be the dihedral angle between *a* and *b*. The generated pieces from such faces present overlapping sections with respect to two-step neighboring pieces due to their shape elongations caused by the additional tilting angle value provided by θ_e. A practical solution for this problem is the truncation of the generated pieces until no overlapping sections are found between them. Still, the truncated pieces must keep the contact interface between them such that the topological interlocking principle holds, and truncating the pieces more than required might eliminate such principle. Figure 9 shows both a TIC and the truncated pieces based on a hinge mesh with $\theta_e = 3\pi / 4$. Meshes with non-planar neighboring faces are common for computational purposes such as 3D modeling, data visualization, and finite element analysis. Considering such meshes is important for the generation of valid TICs.

Height-bisection method

As described in the previous section, a problem during a TIC generation process is setting the right tilting angle for each edge. A slow solution for the problem would be to define first a seed angle value for every edge in the

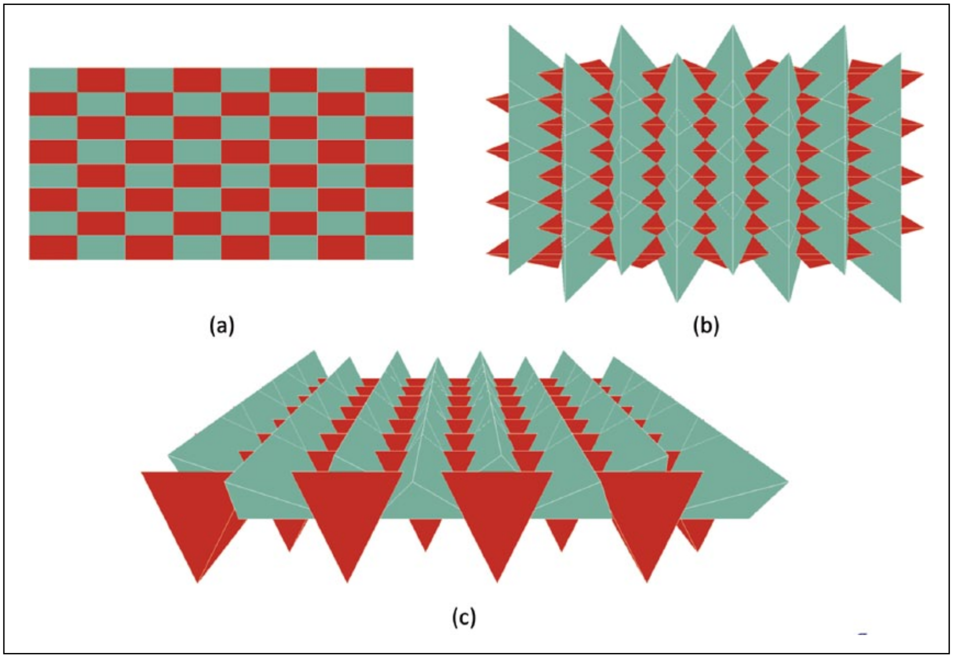


Figure 7. TIC based on a rectangular mesh using the TAM: (a) initial mesh, (b) obtained tetrahedra with $\alpha = \pi/3$, and (c) oblique visualization.

mesh. Then, a TIC is generated and tested for correctness. If overlap is detected, the conflicting edge angle values would be suitably adjusted. These steps would be repeated until the resultant configuration is valid. Adjusting the angles requires the traversal of all the edges in the mesh to recalculate their angle values.

The polyhedra generated using both a checkerboard and the TAM constitute a structure of significant interest. Dyskin et al.[7] reported that both top and bottom sections of the pieces are a grid of squares. When the tessellation squares of the midplane move toward the top and bottom sections, the squares are said to evolve. The vertices of the grid correspond to the projection of the vertices of the generated tetrahedra over the checkerboard. Similar grids are obtained from valid TICs based on meshes with quadrilateral tiles and generated using the TAM. From a geometrical perspective, such a grid is a structure that contains the connectivity information between the faces of the mesh. Each face $f \in F$ is represented by a point p_f, a line segment between points p_a and p_b is defined if there exists an edge *e* such that faces $a, b \in F$ are incident to it. This is the classical representation of the dual graph of a mesh, and infinite line segments are omitted. The projection of the vertices from the squared grid over the checkerboard represents the centers of the tiles; such centers lie at the same location as the respective barycenter, centroid, and the diagonals of each tile. However, the mentioned centers do not lie at the same location for all polygons (and could even be nonexistent as for irregular polygons and their intersecting diagonals). In a technical sense, a center could be any point within the face that does not lie on an edge. Then, any center can be used for building a grid if its calculation can be applied to all faces in the mesh.

The dual graph of a mesh is then a structure that anticipates the orientation of both top and bottom sections of a valid TIC based on such mesh. For a face $f \in F$ with center C_f and normal vector N_f, its respective top and bottom section points T_f and B_f are defined as $T_f = C_f + \lambda N_f$ and $B_f = C_f - \lambda N_f$ for some scalar $\lambda \in \mathbb{R}, \lambda > 0$. Let $a, b \in F$ be two faces of the mesh incident at edge *e*, there exist four planes P_1, P_2, P_3, and P_4 that pass through the edge *e* and contain section points T_a, B_a, T_b, and B_b, respectively.

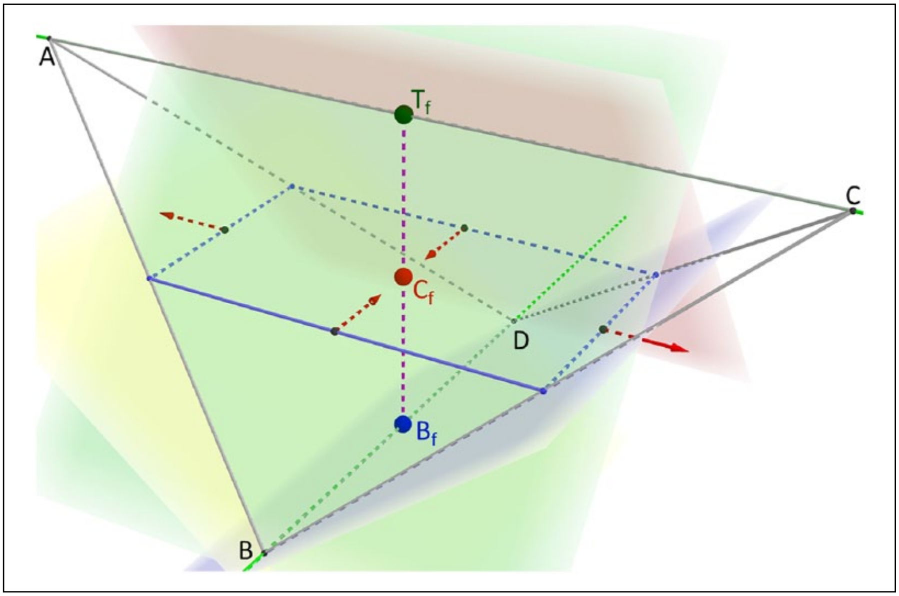


Figure 8. Valid tetrahedron from a rectangular face based on the distance from C_f to both T_f and B_f.

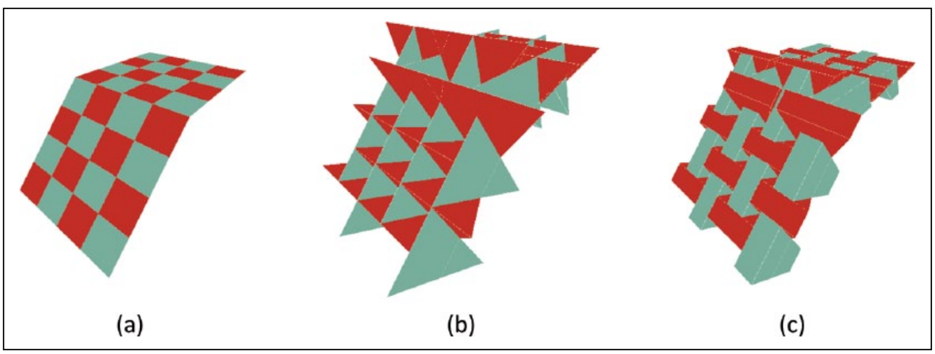


Figure 9. TICs based on a hinge mesh, generated using the TAM with $\alpha = \pi/3$ and $\theta_e = 3\pi/4$: (a) initial mesh, (b) obtained tetrahedra, and (c) truncated tetrahedra until overlapping sections are removed.

Figure 10 shows an example of the four planes incident to the edge, each one of them containing its respective section point. For meshes with regular faces, the planes P_1 and P_4 are the same since section points T_a, B_b, and edge e are coplanar; the same situation occurs with planes P_2 and P_3 and their respective section points. Let e_d be the direction value of edge e as described in the previous section, and e_d indicates which are the planes of interest associated with the angle value of the edge. When e_d goes inward with respect to face a (which is equivalent of going outward with respect to face b), then the planes of interest are P_1 and P_4. However, when e_d goes outward with respect to face a (which is equivalent to going inward with respect to face b), then the planes of interest are P_2 and P_3. The bisector of the dihedral angle between such planes is the tilting angle value associated with edge e required for the generation of a TIC based on the mesh. Then, the angle value for each edge in the mesh is a function of the selected center C_f and λ, which is the distance from C_f to the respective top and bottom section points.

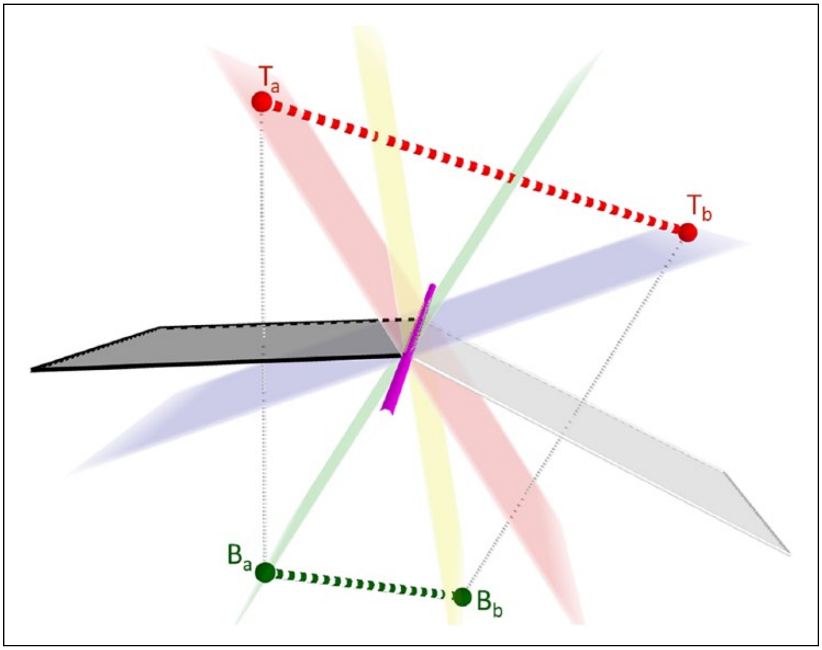


Figure 10. Four planes incident to an edge, each one containing the respective section point. Black quadrilateral corresponds to face *a*, and white quadrilateral corresponds to face *b*. Red line segment represents the connection between top section points T_a and T_b. Green line segment represents the connection between bottom section points B_a and B_b. Red plane contains T_a, green plane contains B_a, blue plane contains T_b, and yellow plane contains B_b. Planes are incident to the edge described in magenta (edge is extended for visualization purposes only).

We propose an algorithm for the angle setup problem for the case of irregular planar meshes based on the previous observations. Let $\text{DCEL}(M)$ and d be defined as mentioned for the TAM. Select the center point C to be used for all faces and set the height value $\lambda > 0$, which is the distance $\text{dist}(C_f, T_f)$ and $\text{dist}(C_f, B_f)$ for each face $f \in \text{faces}$. For each face define: $\text{face}[f].\text{center} = \text{center}(f, C)$, where $\text{center}(f, C)$ is a function that receives a pointer f to the face and the center type C and returns the coordinates of the requested center point. The top and bottom section points associated with each face are defined as

$$\text{faces}[f].\text{top} = \text{faces}[f].\text{center} + \lambda N_f$$
$$\text{faces}[f].\text{bottom} = \text{faces}[f].\text{center} - \lambda N_f$$

where N_f is the normalized normal of the face f. For each halfedge h in the face with an existing twin halfedge set

$$A = \text{halfedges}[h].\text{start}$$
$$B = \text{halfedges}[h].\text{end}$$
$$D = \begin{cases} \text{faces}[f].\text{bottom} & ,\text{halfedges}[h].\text{direction} == 1 \\ \text{faces}[f].\text{top} & ,\text{halfedges}[h].\text{direction} == -1 \end{cases}$$
$$E = \begin{cases} \text{halfedges}[h].\text{twin.face.top} & ,\text{halfedges}[h].\text{direction} == 1 \\ \text{halfedges}[h].\text{twin.face.bottom} & ,\text{halfedges}[h].\text{direction} == -1 \end{cases}$$

that is, points *A, B, D*, and *E* define the planes of interest $P_1 = \text{plane}(A,B,D)$ and $P_2 = \text{plane}(A,B,E)$. Let $N_1 = \|DA \text{ x } DB\|$ and $N_2 = \|EB \text{ x } EA\|$ be the normal vectors of the respective planes of interest. The required bisector plane is represented by its normal vector

$$\text{halfedges}[h].\text{vector} = \|\frac{1}{2}\ (N_1 + N_2)\|$$

Since the direction vector is associated with an edge, it is valid to set $\text{halfedges}[h].\text{twin.vector} = \text{halfedges}[h].\text{vector}$. The vertices of the resultant interlocking block are defined by the intersection of planes associated with three consecutive halfedges in a face as described for the TAM. Figure 11 shows the elements, from two neighboring squares, used by height-bisection method (HBM) for generating the respective interlocking pieces.

The HBM generates the pieces of the configuration using two parameters: a height value and a center of the faces; the actual tilting angle for each edge is not calculated. The selection of a face center point opens the door to slightly different but congruent TICs given the same mesh and height value, as previously mentioned, the required condition for such center point is to be calculated in the same way for all faces of the mesh. The actual angle value (as required in the TAM) can be calculated as the angle between the tilted vector and the normal vector of the edge. Figure 12 shows the TIC generated using the HBM on a rectangular mesh (the same used in Figure 7).

An immediate consequence of the proposed method is the non-generation of pieces from faces next to the outer face of the mesh. That is, no piece is generated from a face where at least one of its halfedges has no twin halfedge or the incident face of the twin halfedge is non-existent. No piece is generated in such situation due to the missing geometric information required for the calculation of the tilting vector of the respective edge.

Results and analysis

Two different TIC generation methods have been described: TAM based on tilting angle values, and HBM based on height values and center choice. The former is traditional for handling the piece generation process, the latter is an improvement for calculating the expected rotation vectors based on both top and bottom section grids. The results to be presented, here and in Appendix 1, compare the two TIC methods for a variety of tessellations in the plane. We also consider TICs based on cylindrical, spherical, and toroidal tessellations.

Results for 2D meshes

When generating a TIC, there are two characteristics:

1. The equatorial midsection of each piece lies in the same plane as the face of the tessellation mesh;
2. On regular planar meshes, the pieces are aligned to each other using different tilting angles for their edges.

Multiple TICs were generated using the generation methods described before, on meshes based of simple polygons further subdivided using midpoint subdivision.[17] In such meshes, the face topology is quadrilateral. Table 2 shows the resultant configurations starting with a subdivided square. This shape is used as the basic case where valid TICs are possible using any parameter values for both generation methods. The resultant configurations preserve the alignment of the pieces, that is, the equatorial section of the pieces exists in the same plane as their respective faces. As expected, the HBM does not generate pieces from the faces in the boundary of the mesh.

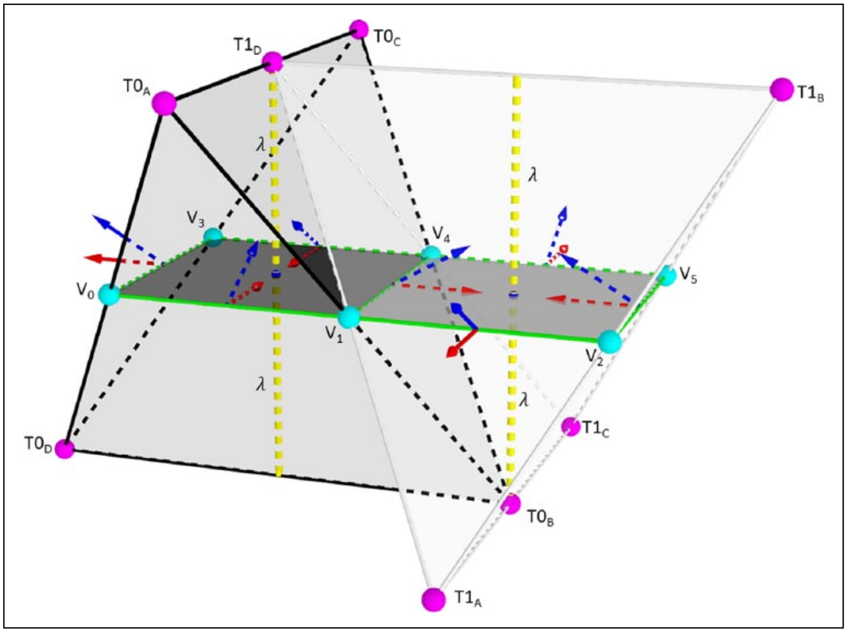


Figure 11. Elements used by HBM for generating the pieces. Cyan points are the vertices of two squares in the tessellation. Red arrows describe the directions associated with the edges of the squares. Yellow dashed lines are the line segments of length λ representing the distance from a face center point to the respective top and bottom sections. Blue arrows are the rotated vectors that describe the tilted planes. Magenta points are the vertices of the resultant interlocking pieces T0 and T1 from left and right squares, respectively.

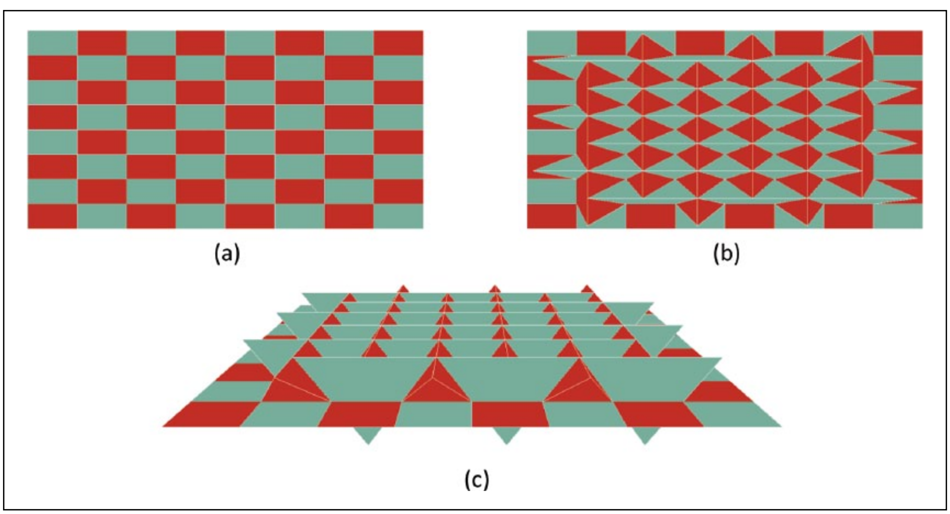
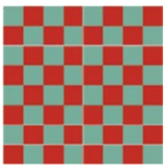
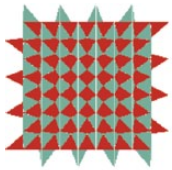
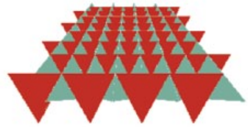
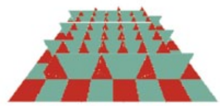


Figure 12. TIC on a rectangular mesh using the HBM: (a) initial mesh, (b) obtained tetrahedra with $h = 0.2$, and (c) oblique visualization.

Table 2. Generated TICs starting from a square with three subdivision iterations. First row: TAM. Second row: HBM.

Original	TIC	Oblique	Alignment
			
			

TIC: topological interlocking configuration; TAM: tilting angle method; HBM: height-bisection method.

Using a different initial shape affects the resultant configurations significantly. Figure 7 shows the generated TIC using the TAM based on a subdivided rectangle. The resultant pieces present overlapping sections around the vertices. This situation is caused by the same angle value for all edges in the mesh. In addition, the alignment of the pieces is not preserved, both top and bottom sections of the pieces do not represent the grid showing the evolution of their midsection toward them. However, the HBM generates a TIC with no overlapping sections and with full alignment between pieces. The generated configuration is shown in Figure 12. Both top and bottom sections of the pieces have the expected evolution grid. Only the pieces from the faces at the boundary are incorrect due to the missing geometric information. Other quadrilaterals were considered for generating TICs using both methods, and Tables 4 and 5 in Appendix 1 show the configurations based on a parallelogram and a trapezoid, respectively. The observations made from the configuration based on a subdivided rectangle apply as well for such quadrilaterals.

The generation methods can be applied to other convex polygons if they are pre-processed by subdivisions. Table 3 shows the resultant TICs based on a subdivided triangle. As expected, the TAM generates pieces that overlap at the vertices; furthermore, the pieces are not aligned with each other. However, the HBM generates pieces with reduced overlapping. The same observations are applied to the generated configurations based on subdivided pentagon and hexagon shown in Tables 6 and 7, respectively, in Appendix 1.

Results on 3D meshes

A major goal of TIC generation is to produce valid configurations for any given 3D mesh. While the 2D case must deal with open meshes at the boundaries, this does not have to be the case in 3D where geometric domains could be partly open (e.g. cylinders and paraboloids) or fully closed (e.g. solids, spheres, and tori). The cylinder (Figure 13) is the first 3D shape into consideration. Since a checkerboard can be mapped onto its surface, both generation methods can be used and are intuitive. Our experiments indicate that the TAM generates pieces with large significant overlap and misalignment. However, the HBM reduces the amount of overlap to small sections between longitudinal pieces only, leaving meridian pieces valid. The resultant configuration shows that the equatorial sections of the pieces lay on the same plane as their respective faces of the mesh.

Table 3. Generated TICs starting from a triangle with three subdivision iterations. First row: TAM. Second row: HBM.

Original	TIC	Oblique	Alignment
			
			

TIC: topological interlocking configuration; TAM: tilting angle method; HBM: height-bisection method.

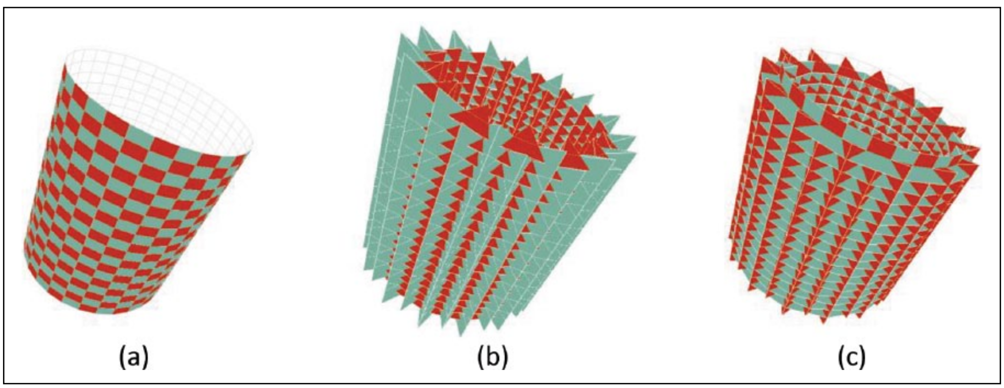


Figure 13. TICs on a cylindrical mesh: (a) cylindrical mesh with radius 2 and height 5, (b) TIC using the TAM with $\theta_e = \pi / 3$, and (c) TIC using the HBM with $h = 1$ and $C_t = \text{barycenter}$.

Spherical surfaces are approximated using inflated subdivided solids. Any Platonic solid approximates a sphere when its faces are subdivided and the distance from the centroid of the solid to each one of its vertices is normalized. Figure 14 shows a sphere approximated by an inflated dodecahedron that has been subdivided three times. The resultant configuration using the TAM generates bumps along the surface which can be traced back to the original faces of the solid, and such bumps are caused by piece misalignment since the same tilting angle is used for all pieces. However, after truncating the resultant pieces, the configuration resembles a nexorade as described in Brocato and Mondardin.[2] Such result suggests there could exist a framework for converting one problem representation to the other given that both concepts are based on Abeille's bond of tetrahedron-based pieces. Finally, when using the HBM, the spherical surface is preserved by the resultant configuration. This is an expected result based on the alignment of the pieces with respect to the faces of the mesh.

We have considered toroidal meshes. Those meshes have both positive and negative curvature faces. Figure 15 shows a torus mesh and the resultant TICs using both generation methods. As with the previous 3D meshes, the TAM does not guarantee the alignment of the pieces and create overlapping pieces. The HBM reduces such overlapping due to the piece alignment. Furthermore, the resultant configuration using the HBM resembles the original mesh more accurately when compared with the configuration generated using the TAM.

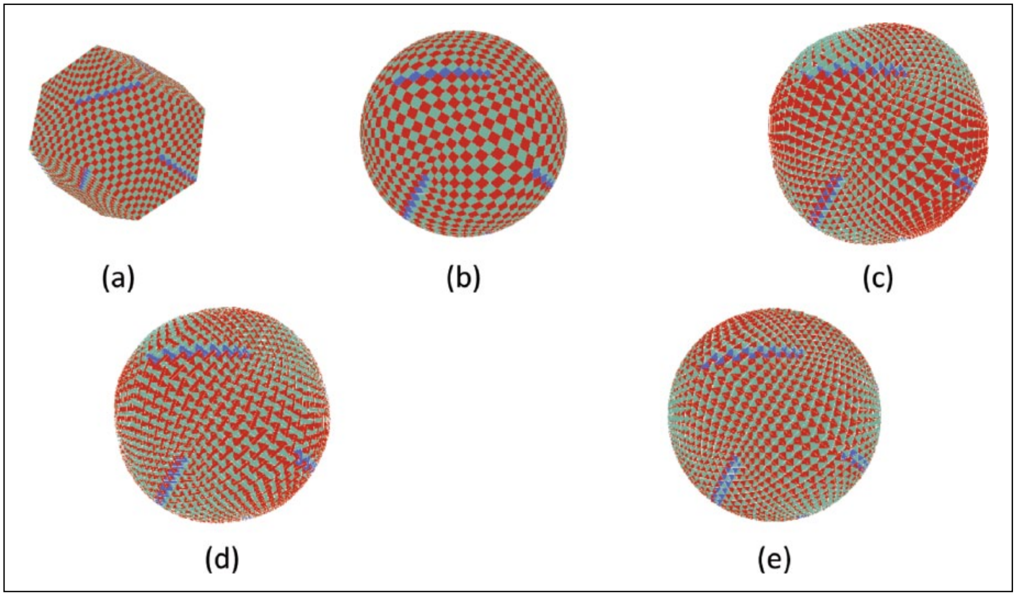


Figure 14. TICs on a spherical mesh: (a) subdivided dodecahedron, (b) inflated dodecahedron (normalized vertices), (c) TIC using the TAM with $\theta_e = \pi / 3$, (d) TIC using the TAM with $\theta_e = \pi / 3$ with truncated pieces, and (e) TIC using the HBM with $h = 1$ and $C_t =$ barycenter. Colors have no meaning, but differentiate adjacent tiles and pieces.

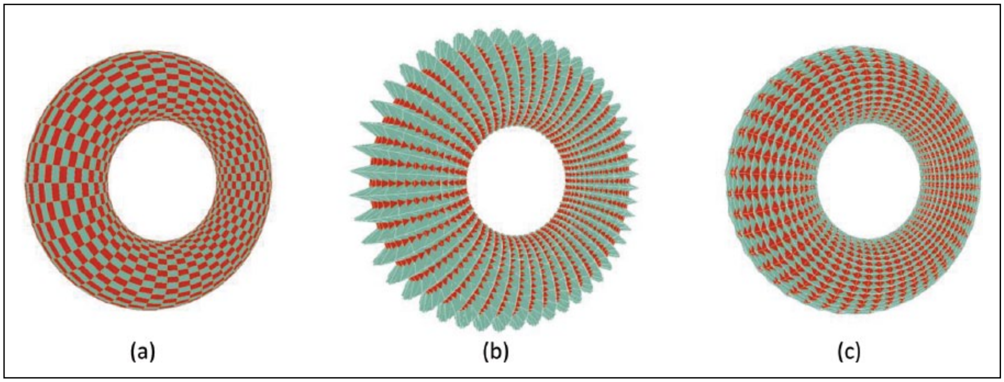


Figure 15. TICs on a torus mesh: (a) torus with external radius 3 and internal radius 1, (b) TIC using the TAM with $\theta_e = \pi / 3$, and (c) TIC using the HBM with $h = 1$ and $C_t =$ barycenter .

Discussion

The HBM is presented as a TIC generation method that preserves the alignment of the top and bottom sections of the pieces as described by the evolution grids of the mesh. The method requires a height parameter λ and a center C of the faces, and the rotation vectors for the edge are calculated in a single iteration over the geometric domain, an improvement with respect to TAM. The resultant configurations maintain the piece alignment, that is, the equatorial sections of the pieces exist on the same plane as their respective faces in the mesh.

Both generation methods were tested on 2D and 3D geometric domains. The generated TICs suggest that the HBM produces configuration with less overlapping than the configurations generated using the TAM. Furthermore, the overlapping sections can be removed by applying a slight truncation over the sections of the pieces. In some cases, the overlapping sections are non-existent due to the correct tilting vector calculated by the HBM. Finding a valid TIC for a 3D mesh increases the complexity of the solution due to the additional information given in the geometric domain. The dihedral angle between incident faces introduces an additional rotation angle that must be considered while determining the location of the top and bottom section points for each face. It could be the case that such section points are not suitable for generating an interlocking piece. Even though piece truncation seems to be the simplest solution, it involves the possibility of invalidating the piece (and even the entire TIC) when the cut nullifies the interface between two neighboring pieces. In terms of shape detail, it was found an increase in the midpoint subdivisions among the faces is required for the resultant TIC to resemble the original domain; however, the trade-off for increasing the level of detail is an increase in the number of calculations to be performed in a single iteration.

Conclusion

The generation of a TIC based on a geometric domain (represented as a surface tessellation) requires designing a piece with interlocking properties from each face. TAM generates the pieces using an angle value as input parameter, and the resultant configuration is valid if the tessellation is composed of regular faces in a planar surface. Otherwise, more than one angle values are required; usually, such angles are calculated on a trial-and-error basis. HBM generates the pieces using a height value and a selected center point for each face as input parameters; the method calculates appropriate rotated vectors that describe the planes incident to the edges of the face. Such rotated vectors are aligned with respect to the top and bottom section grids at a height distance from the geometric domain. HBM generates the configuration in a single iteration over the geometric domain, and the equivalent angle values are intrinsic to the resultant rotated vectors.

HBM reduces the amount of overlapping sections between pieces generated from non-planar geometric domains. Although truncating the pieces properly removes the issue, our interest is to find the right parameter values that avoid overlapping or reduces it to a minimum. We continue the research by including geometric features (e.g. curvature and tessellation type) as elements of the generation methods that help to generate valid configurations from any geometric domain.

Declaration of conflicting interests

The author(s) declared no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

Funding

The author(s) disclosed receipt of the following financial support for the research, authorship, and/or publication of this article: Andres Bejarano was supported in part by NSF award 1361783.

ORCID iD

Andres Bejarano https://orcid.org/0000-0003-2611-2855

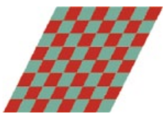
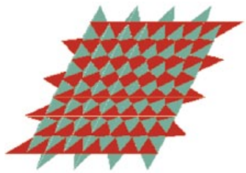
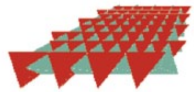
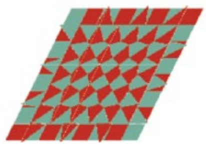
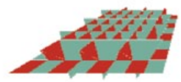
References

1. Fallacara G. Digital stereotomy and topological transformations: reasoning about shape building. In: *Second international congress on construction history*, Queen's College Cambridge, Cambridge, 29 March–2 April 2006.
2. Brocato M and Mondardini L. A new type of stone dome based on Abeille's bond. *Int J Solids Struct* 2012; 49(13): 1786–1801.

3. Brocato M and Mondardini L. Geometric methods and computational mechanics for the design of stone domes based on Abeille's bond. In: Ceccato C, Pauly M, Wallner J, et al. (eds) *Advances in architectural geometry 2010*. New York: Springer, 2010, pp. 149–162.
4. Brocato M, Deleporte W, Mondardini L, et al. A proposal for a new type of prefabricated stone wall. *Int J Space Struct* 2014; 29(2): 97–112.
5. Glickman M. The G-block system of vertically interlocking paving. In: *Proceedings of the 2nd international conference on concrete block paving*, Delft, April 1984.
6. Dyskin A, Estrin Y, Kanel-Belov A, et al. A new concept in design of materials and structures: assemblies of interlocked tetrahedron-shaped elements. *Scripta Materialia* 2001; 44(12): 2689–2694.
7. Dyskin A, Estrin Y, Kanel-Belov A, et al. Toughening by fragmentation–how topology helps. *Adv Eng Mater* 2001; 3(11): 885–888.
8. Dyskin A, Extrin Y, Kanel-Belov A, et al. A new class of composite materials based on topological interlocking. In: *Proceedings of the applied mechanics, progress and applications*, Sydney, NSW, Australia, 20–22 February 2002, pp. 485–490. World Scientific. DOI: 10.1142/9789812777973_0078.
9. Dyskin A, Estrin Y, Kanel-Belov A, et al. Topological interlocking of platonic solids: a way to new materials and structures. *Phil Mag Lett* 2003; 83(3): 197–203.
10. Kanel-Belov A, Dyskin A, Estrin Y, et al. Interlocking of convex polyhedra: towards a geometric theory of fragmented solids. *arXiv*, https://arxiv.org/abs/0812.5089
11. Tessmann O. Topological interlocking assemblies. In: *Proceedings of the 30th eCAADe conference*, Czech Technical University, Prague, 12–14 September 2012.
12. Weizmann M, Amir O and Grobman YJ. Topological interlocking in buildings: a case for the design and construction of floors. *Automat Construct* 2016; 72(1): 18–25.
13. Weizmann M, Amir O and Grobman Y. Topological interlocking in architectural design. In: *Proceedings of the 20th international conference of the association for computer-aided architectural design research in Asia CAADRIA 2015* (Emerging Experience in Past, Present and Future of Digital Architecture), Hong Kong, China, 18–23 May 2015.
14. Weizmann M, Amir O and Grobman YJ. Topological interlocking in architecture: a new design method and computational tool for designing building floors. *Int J Architect Comp* 2017; 15(2): 107–118.
15. Wang Z, Song P and Pauly M. DESIA: a general framework for designing interlocking assemblies. In: *SIGGRAPH Asia 2018 technical papers*, 2018, p. 191, https://lgg.epfl.ch/publications/2018/DESIA/index.php
16. De Berg M, Cheong O, van Kreveld M, et al. *Computational geometry algorithms and applications*. London: Springer, 2008.
17. Armstrong C, Tam T, Robinson D, et al. Automatic generation of well structured meshes using medial axis and surface subdivision. *Adv Design Automat* 1991; 32: 136–146.

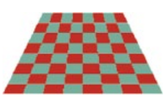
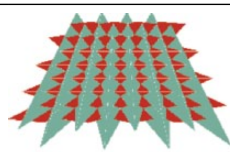
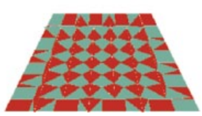
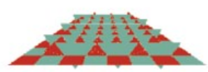
Appendix 1

Table 4. Generated TICs starting from a parallelogram with three subdivision iterations. First row: TAM. Second row: HBM.

Original	TIC	Oblique	Alignment
			
			

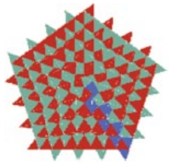
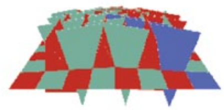
TIC: topological interlocking configuration; TAM: tilting angle method; HBM: height-bisection method.

Table 5. Generated TICs starting from a trapezoid with three subdivision iterations. First row: TAM. Second row: HBM.

Original	TIC	Oblique	Alignment
			
			

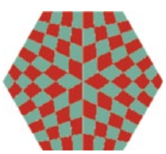
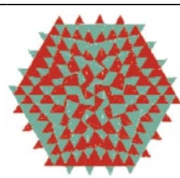
TIC: topological interlocking configuration; TAM: tilting angle method; HBM: height-bisection method.

Table 6. Generated TICs starting from a pentagon with three subdivision iterations. First row: TAM. Second row: HBM.

Original	TIC	Oblique	Alignment
			
			

TIC: topological interlocking configuration; TAM: tilting angle method; HBM: height-bisection method.

Table 7. Generated TICs starting from a hexagon with three subdivision iterations. First row: TAM. Second row: HBM.

Original	TIC	Oblique	Alignment
			
			

TIC: topological interlocking configuration; TAM: tilting angle method; HBM: height-bisection method.