DNA tile self-assembly for 3D-surfaces and genus identification
Florent Becker and Shahrzad Heydarshahi
2023-03-06

Abstract

We introduce a new DNA tile self-assembly model: the Surface Flexible Tile Assembly Model (SFTAM), where 2D tiles are placed on host 3D surfaces made of axis-parallel unit cubes glued together by their faces, called polycubes. The bonds are flexible, so that the assembly can bind on the edges of the polycube. We are interested in the study of SFTAM self-assemblies on 3D surfaces which are not always embeddable in the Euclidean plane, in order to compare their different behaviors and to compute the topological properties of the host surfaces.

We focus on a family of polycubes called cuboids. Order-0 cuboids are polycubes that have six rectangular faces, and order-1 cuboids are made from two order-0 cuboids by subtracting one from the other. Thus, order-1 cuboids can be of genus 0 or of genus 1 (then they contain a tunnel). We are interested in the genus of these structures, and we present a SFTAM tile assembly system that determines the genus of a given order-1 cuboid. The SFTAM tile assembly system which we design, contains a specific set $Y$ of tile types with the following properties. If the assembly is made on a host order-1 cuboid $C$ of genus 0, no tile of $Y$ appears in any producible assembly, but if $C$ has genus 1, every terminal assembly contains at least one tile of $Y$.

Thus, we are able to distinguish the host surfaces according to their genus, by the tiles used in the assembly. This system is specific to order-1 cuboids but the techniques we use should be generalizable to other families of shapes.

1 Introduction

In this paper, we introduce a new tile self-assembly model in order to perform self-assembly on 3-dimensional surfaces. The field of tile self-assembly originsates in the work of Wang [14], who introduced in 1961 Wang tiles, that is, equally sized 2-dimensional unit squares with labels/colors on each edge (later called glues) and designed a Turing universal computation model based on these tiles. In 1998, inspired by Wang tiles and DNA complexes from Seeman’s laboratory [6], Winfree introduced in his PhD thesis [15] the abstract Tile Assembly Model (aTAM). This model uses Wang tiling with an extra information: he associated a non-negative integer strength for each glue label. The power of DNA self-assembly enables to compute using this model. We refer to the survey [9] for more details on the literature, and to the online bibliography of Seeman’s laboratory [13].

Most of the early work in the DNA tile self-assembly literature deals with rigid assemblies in the Euclidean plane $\mathbb{R}^2$ (since the assemblies are discrete, the Euclidean plane is usually seen as the lattice $\mathbb{Z}^2$), which is a natural and simple setting for this model. However, it can be interesting to use self-assembly in richer settings. This has been investigated experimentally for instance in [12, 16, 17] where the assembly takes place on preexisting surface and changes according to the surface. On the theoretical side, there have been some recent works on DNA tile self-assembly outside the Euclidean plane, such as tile self-assembly in mazes [4], where the tile placement is done on the walls of a certain maze. Other types of self-assembly exist that also do not use the Euclidean plane, for example a model of cross-shaped origami tiles [18]. Another type of self-assembly not in the plane is 3D assemblies of complex molecules like crystals [3, 8]. Inspired by this, a recent model called Flexible Tile Assembly Model (FTAM) was introduced in [5], as an extension of earlier work [7]. Here, we have Wang tiles but they self-assemble (without an input surface) in 3D space (modeled by the lattice $\mathbb{Z}^3$) as they can have, in addition of standard rigid bonds, flexible bonds to attach at any angle along the tile edges. The goal of this model is to construct complex 3D structures called polycubes (3D shapes made of unit cubes) [2].

In 2010 Abel et al. used a variant of the aTAM to implement shape replication, where tiles react to the shape of a preexisting pattern to reproduce it [1]. This setting is close to our work, but on the 1
dimensional border of a 2D pattern instead of the 2D surface of 3D objects.

We are interested in studying what happens if we put the tiles on a given 3D surface that is not necessarily topologically equivalent to the Euclidean plane. The intuition is that this could modify the computational behaviour of the tile self-assembly model, and we believe it will be interesting for practical systems.

Inspired by the FTAM, we introduce a new model, called Surface Flexible Tile Assembly Model (SFTAM). In the SFTAM we are given a 3D surface, on which the tiles of the self-assembly get placed. The SFTAM is an intermediate between aTAM and FTAM. Unlike in the FTAM, our aim for introducing the SFTAM is not for building 3D structures or surfaces: we assume the host surface already exists. In the SFTAM, tile bonds are all flexible and the tiles can also bind over the edges of the surface.

This model enables to use self-assembly on surfaces other than $\mathbb{Z}^2$. The aim of this article is to introduce the SFTAM model, and to demonstrate its usefulness by showing how it can be used on various types of surfaces. One of the most important properties of a surface is its genus, which, intuitively, is the number of “holes” in the surface. The Euclidean plane has genus 0. We are interested in using the SFTAM on surfaces with different values of genus. For that, we study the problem of characterizing the surface of the assembly, according to its genus, using the SFTAM.

We will focus on a family of 3D surfaces called cuboids, which are special types of polycubes. Polycubes can form complex surfaces, and their genus can be high. We focus on a simple family of polycubes that can have genus 0 or genus 1. More specifically, we define an order-0 cuboid $C_0$ as a polycube which has only six faces. An order-1 cuboid $C_1 = C_0 \setminus C'_0$ is a polycube that is made from the difference of two order-0 cuboids $C'_0$ and $C''_0$. Thus, an order-0 cuboid is a simple surface with genus 0, but an order-1 cuboid can either have genus 0 (potentially with a pit or concavities) or genus 1, if it has a tunnel.

In this paper, we will suppose that the SFTAM self-assembly is performed on the surface of an order-1 cuboid $C$. We design an SFTAM system whose assemblies differ when $C$ is of genus 0 and of genus 1 and thus, can be used to detect the genus of the surface $C$ of the assembly it is used on. The goal of this study is to show that performing self-assembly on surfaces of higher genus can be helpful. We also demonstrate some techniques which may prove useful in characterizing the topological properties of a wide range of surfaces.

A tile assembly system (TAS) in the SFTAM is defined in a natural way as an extension of the aTAM: tile types are made of four glue labels, each has a strength, there is a single seed tile type and a terminal temperature (more formal definitions will be given later). An assembly is a placement of tiles on facets of the surface of the cuboid $C$. Two tiles bind if they are adjacent (i.e. their placements on the surface share an edge) and their glue labels are the same. In particular, edges are flexible and as a result the tiles can be placed on the border of orthogonal faces of $C$. The assembly is producible if it can be obtained by successfully binding tiles, starting from a seed tile. It is terminal if no additional tile can be bound to an existing tile.

Let $C = C_0 \setminus C'_0$ be an order-1 cuboid with its three dimensions at least 10 for $C'_0$. Our main result is to describe an SFTAM (TAS) $S_C$ with a subset $Y$ of its tile types such that the following holds:

- if the order-1 cuboid $C$ has genus 0, then no tile of $Y$ appears in any producible assembly of $S_C$ on $C$, and
- if $C$ has genus 1, every terminal assembly of $S_C$ on $C$ contains at least one tile of $Y$.

In other words, if no single tile from $Y$ appears in all terminal assemblies of $S_C$ on $C$, its genus is equal to 0, but the presence of even a single tile from $Y$ in an assembly shows that the genus of $C$ is 1. Thus, the genus of $C$ can be determined using the assemblies of $S_C$ on $C$. The assemblies of $S_C$ consist of two phases: a skeleton forms on the cuboid and separates it into several regions, then the regions are partially filled by inner tiles. See Fig. 1 for a sketch of the skeleton and its inner filling for an order-1 cuboid with genus 0. When the cuboid has genus 1, we show that there must be some parts of the skeleton or the inner filling which intersect in a way that is not possible on a genus-0 cuboid. The tile types of $Y$ stick at the place where this happens. See Fig. 2.

We start with basic definitions and notations in Section 2, where we introduce and formalize our SFTAM model. Next, we introduce the family of order-1 cuboids and we show how SFTAM behaves on the family of order-1 cuboids as an assembly model in three dimensions. In Section 3, we develop technical lemmas that will be necessary for the proof of our main result. In Section 4, we present our main result: a SFTAM tile assembly system that identifies the genus of order-1 cuboids using specific tiles from that system. We conclude in Section 5.
Figure 1: The skeleton of the assembly on the order-1 cuboids are showed in color, the regions partially fill by \( t_{\text{odd}} \) and \( t_{\text{even}} \) tile types.

(a) The case where the tunnel of an order-1 cuboid is shown by a tile of type \( t_{\text{reg}} \), when the inner filling tiles of two neighbor regions meet each other.

(b) The tunnel intersects along the width of plane \( P_X \) and length of plane \( P_Y \).

(c) The case where the tunnel of an order-1 cuboid is shown by a tile of type \( t_{\text{ibc}} \), located at the intersection of the skeleton.

Figure 2: The cases that the tunnel of an order-1 cuboid are showed by a particular tile type
2 Definitions and notations

In this section, we present the notions and definitions that are used throughout this paper. First, we explain what we mean by a tile assembly model for 3D surfaces in $\mathbb{Z}^3$ and we present our model: the Surface Flexible Tile assembly model or SFTAM. Then we present the family of surfaces that we work on: order-1 cuboids.

2.1 The tile assembly model on surfaces in $\mathbb{Z}^3$: SFTAM

We now define the Surface Flexible Tile assembly Model, SFTAM. We work in 3-dimensional space, on the integer lattice $\mathbb{Z}^3$.

**Definition 1** (Tile type in SFTAM). Let $\Sigma$ be a finite label alphabet and $\epsilon$ represent the null label. A tile type $t$ is a 4-tuple $t = (t_1, t_2, t_3, t_4)$ with $t_i \in \Sigma \cup \{\epsilon\}$ for each $i \in \{1, 2, 3, 4\}$. Each copy of a tile type is a tile.

Tiles are 2D unit squares whose sides are assigned the labels of the tile type. These squares are allowed to translate and rotate (unlike in aTAM), but do not reflect (unlike in FTAM). In fact, since the tiles stick to a given surface, we can assume that they have an inner face and an outer face and that they always attach with the inner face in contact with the surface. In the definition of tile types, we show labels by numbers rather than cardinal directions. However, often, the orientation of a tile dictates the orientation of the tiles around it. Then, we use the expression “northern label” to refer to the label which will end up on the northern side (and similarly for east, west, south).

**Definition 2** (Facet). A facet is a face of the lattice $\mathbb{Z}^3$, i.e. a unit square whose vertices have integer coordinates.

**Definition 3** (Polycube). A polycube is a 3D structure that is a subset of $\mathbb{Z}^3$ and is formed by the union of unit cubes that are attached by their faces.

For a given facet of a polycube, there are four possibilities for placing a given tile. We formalize this as follows.

**Definition 4** (Placement). Let $C$ be a polycube. A placement $p = (f, o)$ on $C$ consists of a facet $f$ on the surface of $C$, and a side $o$ of $f$, called its orientation.

We denote the set of all placements in $C$ by $Pl(C)$.

Given a tile type $t = (t_1, t_2, t_3, t_4)$ and a placement $p = (f, o)$, placing $t$ at the placement $p$ defines a mapping from the edges of $f$ to the glues of $t$ : the $i$-th side of $f$ (starting from the orientation $o$ and going clockwise, looking from the exterior of the surface of $C$) is associated with $t_i$.

Notice that the normal vector $n$ of the placement in the FTAM is not needed in the SFTAM. Indeed, in the SFTAM, the tiles do not reflect. The reason is that we fix that the normal vector (as used in the FTAM) always starts inside of the polycube and points to the outside of the structure. So, the order of the tiles’ sides is uniquely determined by the orientation of its placement.

Now we can define a tile assembly system in the SFTAM.

**Definition 5** (Tile assembly system (TAS) in the SFTAM over a surface). A tile assembly system, or TAS, over the surface of a given polycube $C$ is a quintuple $S = (\Sigma, T, \sigma, str, \tau)$, where:

- $\Sigma$ is an alphabet,
- $T$ is a finite set of tile types on $\Sigma$,
- $\sigma$ is a single one tile or a block of tiles, called seed,
- $str$ is a function from $\Sigma \cup \{\epsilon\}$ to non-negative integers called strength function such that $str(\epsilon) = 0$, and
- $\tau \in \mathbb{N}$ is called the temperature.

Next, we present some further notations. The following definitions of the assembly dynamics are the same as in the aTAM [9].
Definition 6. An assembly $\alpha$ of a SFTAM TAS $S = (\Sigma, T, \sigma, \tau)$ on a polycube $C$ is a partial function $\alpha : \text{Plc}(C) \rightarrow T$ defined on at least one placement and such that for each facet $f$ of $C$, there is at most one placement $(f, \alpha)$ where $\alpha$ is defined.

For placements $p = (f, \alpha)$ and $p' = (f', \alpha')$ of $\text{Plc}(C)$ with $\alpha(p) = t$ and $\alpha(p') = t'$ such that $f$ and $f'$ are distinct but have a common side $s$, we say that $t$ and $t'$ bind if the glue of $t$ and $t'$ placed on $s$ are equal and have positive strength.

The assembly graph $G_{\alpha}$ associated to $\alpha$ has as its vertices, the placements of $\text{Plc}(C)$ that have an image by $\alpha$, and two placements $p$ and $p'$ are adjacent in $G_{\alpha}$ if the tiles $\alpha(p)$ and $\alpha(p')$ bind.

An assembly $\alpha$ is $\tau$-stable if for each cut $c$ of $G_{\alpha}$, the sum of the strengths of edges of $c$ with complementary labels is at least $\tau$.

We start with a seed and then we add tiles one by one. This is formally described as follows.

Definition 7. Let $C$ be a polycube and $S = (T, \sigma, \tau)$ a SFTAM TAS with $\sigma$ positioned on a placement of $C$. An assembly $\alpha$ of $S$ is producible on $C$ if either $\alpha$ is $\alpha : f \mapsto \sigma$ where $f$ is a facet of $C$, or if $\alpha$ can be obtained from a producible assembly $\beta$ by $\tau$-stably adding a single tile from $T \setminus \sigma$ on $C$.

We write $\beta \rightarrow^\tau \alpha$ when $\alpha$ is producible from $\beta$ and we denote the set of producible assemblies of $S$ by $A^\tau[S]$.

An assembly is terminal if no tile can be $\tau$-stably attached on $C$. The set of producible, terminal assemblies of $S$ is denoted by $A^\tau_T[S]$.

2.2 Cuboids

In this part, we introduce the structures that we work on: order-1 cuboids, which are special types of polycubes. We work in 3-dimensional space, on the integer lattice $\mathbb{Z}^3$. We start with some definitions.

Definition 8 (Order-0 cuboid). An order-0 cuboid $C = (s_C, x_C, y_C, z_C)$ where $s_C = (s_x, s_y, s_z) \in \mathbb{Z}^3$ is the point of $C$ with smallest coordinates and $x_C, y_C, z_C$ are integers representing the length, width and height of $C$ is a 3D structure containing all points $(x, y, z)$ of $\mathbb{Z}^3$ such that $s_x \leq x \leq s_x + x_C$, $s_y \leq y \leq s_y + y_C$ and $s_z \leq z \leq s_z + z_C$. We denote the set of all cuboids by $O_0$.

Note that we work on the boundary surface of cuboids.

We are interested in 3D structures that are more complicated than order-0 cuboids, in particular 3D structures that can have tunnels, that is, “holes”. To this aim, we introduce a family of polycubes called order-1 cuboids.

Definition 9 (Order-1 cuboid). An order-1 cuboid $C_1$ is the difference between two elements of $O_0$. Given $C_0 = (s_{C_0}, x_{C_0}, y_{C_0}, z_{C_0})$ and $C'_{0} = (s_{C'_{0}}, x_{C'_{0}}, y_{C'_{0}}, z_{C'_{0}})$ in $O_0$. If for $i \in \{x, y, z\}$ there is a $i_{C_0} \leq i_{C'_{0}}$ such that $C_{1} = C_0 \setminus C'_{0}$, is an order-1 cuboid. We note $O_1$ the set of all order-1 cuboids.

Note that an order-0 cuboid is a classic “cuboid”, i.e. it has six rectangular faces. An order-1 cuboid can have an asymmetric surface, including a hole or a concavity. Moreover, the condition on the dimension of $C_0$ and $C'_{0}$ ensure that the surface of $C_1$ is connected.

The genus of an order-1 cuboid is at most 1. Note that the set of order-0 cuboids is a subset of the set of order-1 cuboids, that is, $O_0 \subseteq O_1$. Thus, an order-1 cuboid $C_1 = C_0 \setminus C'_{0}$ can be of three different types, depending on how $C_0$ and $C'_{0}$ interact: (i) $C_0$ and $C'_{0}$ have no intersection, and $C_1$ is an order-0 cuboid, (ii) $C'_{0}$ cuts a hole in $C_0$, and $C_1$ is named as an order-1 cuboid with a tunnel and has genus 1; and (iii) $C_0$ and $C'_{0}$ intersect but $C'_{0}$ does not cut a hole in $C_0$. If the cut is in the inner of a face of $C_0$, $C_1$ is an order-1 cuboid with a pit and its genus is 0. If the cut is on a corner of $C_0$, then $C_1$ is an order-1 cuboid with a concavity and its genus is 0.

We denote by $O_1^T$ the set of order-1 cuboids that are not order-0 cuboids, that is, the ones from items (ii) and (iii) above.

The set of cuboids of $O_1^T$ with a pit are denoted by $O_1^P$, the ones with a tunnel, by $O_1^T$, and the ones with a concavity, by $O_1^C$. 
3 Technical definitions and lemmas: counters, U-turns and middle finding

Given an assembly on $\mathbb{Z}^2$, we call the smallest axis-parallel rectangle containing the assembly, its underlying rectangle. If an SFTAM assembly is on a 3D surface, it is permitted to fold along the tiles’ edges. It is then possible to generalize the underlying rectangle as the smallest subset of the surface which contains the assembly and is isomorphic to a rectangle of $\mathbb{Z}^2$, if it exists. See Fig. 3 for an illustration. In this section, we present some lemmas about the SFTAM, but for simplicity we assume to be on $\mathbb{Z}^2$. However, due to the following lemma, the results of this section are applicable to assemblies on polycube surfaces as well.

Lemma 10. Let $\alpha$ be a producible assembly of an SFTAM TAS $S$ on $\mathbb{Z}^2$ with underlying rectangle $R$, and let $C$ be a polycube. If there exists a function $i : \mathbb{Z}^2 \to C$ such that the restriction of $i$ to $R$ is a graph isomorphism, then the image of $\alpha$ under $i$ is producible on $C$.

Proof. If the seed of $S$ is placed at $p_s$ in $\mathbb{Z}^2$, it is placed at $i(p_s)$ on $C$. Since the tile bonds can fold along edges of $C$, the assembly on $C$ proceeds exactly as it proceeds on $\mathbb{Z}^2$, and each tile placed at a point $p$ in $\mathbb{Z}^2$ is placed at point $i(p)$ on $C$. \qed

![Figure 3: Finding the middle of a surface. The underlying rectangle is in dark gray.](image)

We introduce two increasing and decreasing counter TASs. Then, by combining a TAS called U-turn system with the counter systems, we establish a lemma called middle finding system lemma to find the middle length of a given surface. This lemma paves the way to the proof of our main result.

There is another essential point before starting: we will represent a number via tiles, in a way that is classic in the literature of tile self assembly, see for example [11]. To this aim, we use the binary representation of numbers and we associate a number to a row of tiles such that every bit of the number is assigned to a tile with the appropriate label.

Definition 11 (Row tile number). Let $T_0$ and $T_1$ be two sets of tiles with labels 0 and 1, respectively. Let $N$ be an integer and $a_1a_2a_n$ be its binary representation with $n = \lceil \log_2 N \rceil$. A row of tiles with labels $a_1^*, a_2, ..., a_n$ is the row tile number representation of $N$ such that the distinct tile $a_1^*$, represents the most significant bit of the number.

We see an example of the row tile number of the number 12 in Fig. 4.
3 TECHNICAL DEFINITIONS AND LEMMAS: COUNTERS, U-TURNS AND MIDDLE FINDING

3.1 The increasing binary counter system

In this section, we introduce the Increasing Binary Counter system (IBC). First, we explain what the IBC system is, then we present its tile types. Afterwards, we show how it works by explaining the assembly process. This system is a base for the middle finding system.

A different binary counter system was introduced in [11]. The advantage of our IBC system is that we do not need the support tiles at the bottom of the production. Especially, there is no infinite (or arbitrarily large) line of 0 tiles at the left of the binary number of each row. Thus, the counter does not fill the whole plane and instead we will have a counter ribbon that is able to measure the length of a given support on the side of a surface.

In the production of the IBC system, the tiles of each row (excluding the tiles $\sigma_+$ and $t_{++}$) form a row tile number representing the index of the row in which they are located. Consequently, whenever the assembly stops by getting to the end of the given surface, the length of the assembly corresponds to the last row tile number. This number is a measure of the length of the underlying rectangle of the assembly.

Lemma 12 (The increasing binary counter system, IBC). The increasing binary counter system or IBC system, is a SFTAM tile assembly system over $\mathbb{Z}^2$ such that if the bottom of the seed is placed at the origin, each row (excluding the tiles of type $\sigma_+$ and $t_{++}$) denotes its position in binary on the y axis and the leftmost tiles of all rows whose number is $2^k$ for some $k \geq 1$ is of type $t_O$.

The system is described by a quintuple $S = (\Sigma, T, \sigma_I, str, \tau)$, where:

- $\Sigma = \{0, 1, s, ++, over, \epsilon\}$
- $T = \{\sigma_+, t_{++}, t_{0+0}, t_{0+1}, t_{1+0}, t_{1+1}, t_S, t_O, t_{OS}\}$
- $\sigma_I$ is a block of tiles made of a tile of type $\sigma_+ = (1', \epsilon, \epsilon, 1')$ (the $\epsilon$ labels are arbitrary) together with a column of support tiles placed on the north side of $\sigma_+$, all of whose west labels are 1 (the other labels can be arbitrary).
- $str : \Sigma \cup \{\epsilon\} \rightarrow \{0, 1, 2\}$ such that $str(\epsilon) = 0$. The values of the $str$ function for $\Sigma$ are $str(0) = str(1) = str(++) = str(s) = 1$ and $str(+) = str(over) = 2$.
- $\tau = 2$

The IBC tile types are shown in Fig. 5.

Proof. We start by describing the IBC tiles.

The seed of the IBC.

There is one tile type $\sigma_+ = (++, \epsilon, \epsilon, over)$, and one incrementor tile type $t_{++} = (++, \epsilon, ++, 1)$ (the gray tile in Fig. 5) called the support tiles. Together, one tile of type $\sigma_+$ with a column of tiles of type
t_{++} form the block seed of the IBC. Note that the tile type of the support is allowed to be different from
t_{++}, as long as its western labels are 1.

The rule tile types of IBC.

There are also seven sum rule tile types: four "standard" rule tile types and three rule tile types related
to the most significant bit. However, from now on when we mention rule tiles, we refer to standard rule
tiles.

First, we explain how the binary sum is done by these tiles. In each rule tile type, the inputs are the
labels at the east and south of the tile, for example a and b. The two outputs are: the classic binary sum
a + b in the northern label, and the carry bit, which is displayed on the western label of the tile. The
rule tile types of the IBC are: \( t_{0+0} = (0,0,0,0), t_{0+1} = (1,0,1,1), t_{1+0} = (1,1,0,0) \) and \( t_{1+1} = (0,1,1,0) \)
with the side labels as mentioned, and where the inner label is the value of the sum, the same as the
northern label.

The most significant bit tile types of the IBC.

There are three tile types for the most significant bit. They use the same logic as the rule tile types.
The tile type \( t_S = (1_S,0,1,\epsilon) \) represents the most significant bit of the row number when that number
is not a power of 2. In case \( a = b = 1 \) in the currently most significant bit, there is an overflow after
the sum operation \( a + b \). In this case, the two rule tiles \( t_{OS}, t_O \) are used to change the most significant
bit. The tile \( t_{OS} = (0,1,1,\overline{1}) \) is the connecting tile between \( t_O \) and \( t_S \) in case of the overflow and
\( t_O = (1_S,\overline{1},\overline{1},\epsilon,\epsilon) \) is the new most significant bit when the row number is a power of 2.

The process of the assembly in the IBC TAS.

The process starts with the seed. The tiles of the support start the process of incrementing the row
below.

A tile of type \( t_O \) binds to the east of the seed tile by label "over" with strength 2 in the east. Note
that the label of \( t_O \) is 1, which corresponds to row number 1, where it is located. Now the incrementation
process starts. Each tile is bound by its eastern and southern neighbors, i.e. for having a tile \( t_a \), with the
label \( a + b \), there is a tile at the south with north label \( a \) and there is a tile at the east with west label
(0. The first tile to bind in each row is the easternmost, it binds with the support. The carry bit of the
sum appears in the west side of the tile \( t_a \). The northern label of \( t_a \) is \( a + b \) too. See an example of an
assembly for the number 16 in Fig. 7.

Note again that each row represents the row number in which it is located along the y axis. This
process can continue for the whole height of the support \( \sigma_1 \).

In the terminal assembly, the top row is the row tile number \( N \), where \( N \) is the length of the support.
Moreover, the tiles of type \( t_O \) only appear at the left of a row when the row is a power of 2, and the
highest one is the row number \( \max\{c \mid 2^c \leq N\} \).

3.2 The Decreasing Binary Counter System

The decreasing binary counter system is a tile assembly system that implements a reverse counter. This
process is within the framework of Decreasing binary counter system or DBC system for short, formally
defined as follows.

Lemma 13 (Decreasing Binary Counter system, DBC system). The Decreasing binary counter system
or DBC system for short, is a tile assembly system over \( \mathbb{Z}^2 \). It is defined by a quintuple \( \mathcal{S} = (\Sigma, T, \sigma_D, str, \tau) \),
where:

- \( \Sigma = \{0,0',1,1',-,-,\epsilon\} \)
- \( T = \{\sigma_-, t_-, t_{0-0}, t_{0-1}, t_{1-0}, t_{1-1}, t_{00-0}, t_{00-1}, t_{11-0}, t_{11-1}\} \), see Fig. 7.
- \( \sigma_D \) is a block made of a binary row tile number \( N \), a tile of type \( \sigma_- = (-,-,\epsilon,\epsilon,0) \) (where the \( \epsilon \)
labels are arbitrary) at the east of it, and a column of support tiles that have western label 1 (for
example of type \( t_- \)) placed on the north of the \( \sigma_- \) tile. The column has height at least \( N \).
- \( str : \Sigma \cup \{\epsilon\} \rightarrow \{0,1,2\} \) such that \( str(\epsilon) = 0 \), \( str(-) = 2 \) and all other values are 1.
- \( \tau = 2 \)
Figure 6: Assembly of 16 in the IBC.

The terminal assembly of the DBC system is a rectangle with exactly one tile of type $t_{1-0}^*$ located at the left of the $(N+1)^{st}$ row.

Proof. The DBC tiles are shown in Fig. 7.

The seed of the DBC.

An assembly of the DBC TAS starts when a binary row number binds to the decrement operation tile of type $\sigma_D = (--,\epsilon,\epsilon,0)$ to form a DBC seed for the decrementation operation.

There is a decrementor tile of type $t_{--} = (-1,\epsilon,-1,1)$ which triggers the decrementation of the row below it into a new row to the left of the next tile of type $t_{--}$.

The rule tile types of the DBC.

There are eight rule tiles, four for the most significant bit and four for the others. The inputs are the labels of the east and south of the tile, for example $a$ and $b$, and there are two outputs. The first one,
that is located in the northern label, is the result of the binary subtraction \(a - b\). The second output is the western label of the tile such that its value is needed to borrow for the next operation. The label of the new tile itself is also \(a - b\). With this description, the rule tiles are \(t_{a-b} = (a - b, b, a, \text{borrow})\), i.e. \(t_{0-0} = (0, 0, 0, 0)\), \(t_{0-1} = (0, 1, 1, 1)\), \(t_{1-0} = (1, 0, 1, 0)\), \(t_{1-1} = (1, 1, 0, 0)\), \(t^0_{0-0} = (0', 0, 0', \epsilon)\), \(t^0_{0-1} = (0', 1, 1', \epsilon)\), \(t^1_{0-0} = (1', 0, 1', \epsilon)\), \(t^1_{0-1} = (1', 1, 0', \epsilon)\). See Fig. 9 for an illustration.

The process of the assembly in the DBC TAS.

The counter starts at an arbitrary value \(N\) and grows along a vertical column of support tiles. A decrementation process starts by relying on the support tiles that trigger the decrementation of the row below it into a new row to the left of the support tile. Each row \(i\) shows the number \(N - i\) modulo \(\min\{2^k \mid N \leq 2^k\}\). Indeed, when the row’s value counts down to 0, the value of the next row will be the bits’ maximum possible capacity of the number that the process was started with. Moreover, note that the most significant bit tiles are the tiles of types \(t^0_{0-0}\), \(t^0_{0-1}\), \(t^1_{0-1}\) and the row of the number 0 for the first time is exposed whenever a tile \(t^1_{0-0}\) appears in the most significant bit in the row after the one of 0, i.e. the row of −1. An example of a DBC assembly for \(N = 12\) is presented in Fig. 8 where the tile of type \(t^1_{0-0}\) is presented in red. After the appearance of \(t^1_{0-0}\), the tiles of types \(t^0_{0-0}\), \(t^0_{0-1}\), \(t^1_{0-0}\), \(t^1_{0-1}\) do not appear in the assembly anymore and the process continues with tiles of types \(t_{0-0}\), \(t_{0-1}\), \(t_{1-0}\) and \(t_{1-1}\). After that, the process repeats for the whole length of the support. Since the length of the support is at least \(N\), the terminal assembly is a rectangle with exactly one tile of type \(t^1_{0-0}\) located at the left of the \((N + 1)\)st row.

3.3 The U-turn system

We now introduce a TAS which copies a row tile number to its left by doing a U-shape rotation of each of its tiles. The seed is analogous to that of IBC and DBC. The support must be at least as long as the width of the input.

**Lemma 14** (U-turn system lemma). There is a TAS \(S_U = (\Sigma, T_U, \sigma_U, \text{str}, \tau)\) such that for a given row tile number \(N\), \(S_U\) makes a copy of \(N\) to the left of its most significant bit. More precisely, if \(N\) was in positions \([x, y], \ldots, [x + k, y]\) in the terminal assembly, there is a copy of \(N\) in positions \([x - k - 1, y], \ldots, (x - 1, y)]\).

The system is defined as follows.

- \(\Sigma = \{0, 0', 1, 1', 1^*, a, b, b', c, d, \epsilon\}\)
- \(T_U = \{\sigma_u, t_U, t_b, t_c, t^0_u, t^1_u, t^1_v, t^1_\tau, t^1_\epsilon, t^0_\tau, t^0_v, t^0_\epsilon, t^1_\epsilon\}\).
The block seed \( \sigma_U \) is formed by a tile of type \( \sigma_u = (\epsilon, \epsilon, b'', c) \) (where \( \epsilon \) can be arbitrary labels) which is bound to the right of a row tile number, and to the north of the \( \sigma_u \) tile, a vertical column of support tiles (with at least as many tiles as in the row tile number) with western label \( b \).

- \( \text{str} : \Sigma \cup \{\epsilon\} \rightarrow \{0, 1, 2\} \) such that \( \text{str}(\epsilon) = 0 \). The values of the \( \text{str} \) function in \( \Sigma \) are \( \text{str}(0'') = \text{str}(1'') = \text{str}(b'') = 2 \), and 1 for the others.
Figure 10: Copying 11111001 in the U-turn system. The gray bracket on the right shows the minimum number of support tiles that are necessary for this assembly.

- $\tau = 2$

Proof. Let $T_U$ be the set of tiles shown in Fig. [11], its temperature is 2.

The seed and the block tiles in the U-turn system.

The U-turn system is a TAS that starts with a multiple seed such that the number of support tiles is at least the number of tiles of the row tile number. Let the tile $\sigma_u = (\epsilon, \epsilon, b', c)$ bind to a row tile number $N$ with $n = \lceil \log_2 N \rceil$. We assume that the assembly is oriented as in Fig. [10] where the initial row tile number is at the top right. From the south of $\sigma_u$, support tiles of type $t_u = (b', \epsilon, b'', b)$ with northern and southern labels $b''$ whose strength are 2, are placed one after another. These tiles form a ribbon of tiles of type $t_u$ that is perpendicular with respect to the seed. Once the number of support tiles becomes $n$, all these tiles together form the seed of the U-turn system, and the assembly begins.
After this step, the tiles below the seed start to grow and the assembly continues row by row. The overview of the assembly’s process of the U-turn system is as below.

Overview of the assembly’s process in the U-turn system.

The word “U-turn” refers to the process of the assembly in this system: each bit tile is copied along a U-shaped path to be copied to the left of the initial row tile number. More precisely, each tile $a_k$ that is placed at the $k$-th bit of the row tile number, is copied $k$ times to the south, then it is copied $n$ times to the west, and at the end $k$ times to the north. Furthermore, the assembly progresses row by row under the seed. Indeed, the last tile of each row has strength 2 at the west and increments the number of row tiles. Thus, the next tile is placed at the $(n+k)$-th column (starting from the rightmost column of Fig. 10). This tile makes a base, and using the support of the last tiles of each row, a vertical column is constructed that sends the value of $a_k$ to the north.

The first stage of the assembly in the U-turn system.

First, we explain the tiles of the very first row under the seed. For an illustration during this stage see Fig. 12. This row is made of the tiles that transfer the value of the rightmost tile i.e. the least significant bit $a_1$ of the row tile number, to the column $n+1$. At first, a tile of type $t_0^1 = (0, b, b, 0)$ or $t_1^1 = (1, b, b, 1)$ (depending on the value of the least significant bit, for short we use $t_1^1$) appears in the assembly. The tiles of type $t_1^1$ transfer the value of the north label to the west label. These tiles are colored in dark green in Fig. 10.

Then, tiles of type $t_{1,1}^1 = (1, 1, 1, 1)$, $t_{1,0}^1 = (1, 0, 1, 0)$, $t_{0,1}^1 = (0, 1, 0, 1)$ and $t_{0,0}^1 = (0, 0, 0, 0)$ (for short $t_{1,0}^1$, light green in Fig. 10), that copy the north label to the south label, and the east label to the west label, appear under the row tile number $N$, at the west of the tile of type $t_1$ (dark green in Fig. 10).
except the leftmost one, which corresponds to the most significant bit. Notice that in the U-turn system, all the tile types that transfer the value of the most significant bit tile are distinct from the other tiles.

The tile below the most significant bit is a special tile of type \( t_{1\text{-}n} = (1^s, 1, 1) \) or \( t_{0\text{-}n} = (1^r, 0, 1, 0') \) (cyan in Fig. 10) that copies the north label to the south label and the east label to the west label with strength 2. Thus, this type of tile \( t_{1\text{-}n} \) or \( t_{0\text{-}n} \) allows to open a new column. The tile after it is of type \( t_{0''} = (0', d, c, 0) \) or \( t_{1''} = (1', d, c, 1) \) (light blue in Fig. 10) with strength 2 at the east label and brings its value to the north. Now, the first copied tile, which corresponds to the least significant bit tile, appears at the left of the most significant bit tile of \( N \).

Note that the tile of type \( t_j \) also copies the east label, i.e. block label \( b \) from the support tiles \( t_u \), to the south label. Thus, at the south of the tile of type \( t_j \), the tiles of types \( t_{0\text{-}u} = (b, b, b, b) \) (named block tiles) appear vertically with the support of \( t_u \).

The overall transfer of the value in the U-turn assembly.

Now, we explain the general process of the assembly for transferring the value of the \( k \)-th least significant bit of the row tile number. In Fig. 13 this stage is marked with yellow-filled rectangles on the left side of the assembly. In this part, each new tile can be placed if and only if their northern and eastern labels correspond to the previous tiles.

Assume that we are in the \( k \)-th stage i.e. the value of the \( k \)-th least significant bit is being transferred.

In the previous stages before \( k \), the block tiles of types \( t_{0\text{-}u} \) grow vertically from the south of the tile of type \( t_j \). Thus, the tiles of type \( t_{0\text{-}u} \) are placed between the support tiles of type \( t_u \) and \( t_{j} \) in each row and copy the block label \( b \) from the east to the west. Therefore, the tile \( t_j \) appears in the \( k \)-th row and the \( k \)-th column with the help of the tiles of type \( t_{0} \) in the north and \( t_{0\text{-}u} \) at its east. That being said, the tile that is placed at the \( k \)-th bit of the row number is copied \( k-1 \) times to the south before it meets the block tile, the place where the tile of type \( t_j \) (that transfers the north label to the west label) appears.

Then, the value of the \( k \)-th bit tile is copied one time to the south, and \( n \) times to the west. Before arriving at the \((k+n)\)-th column, the value passes through the \( n \)-th column, i.e. the column of the most significant bit tiles.

As mentioned before, the path that the most significant bit goes through is created with marked tiles to keep it distinguished from the other bit tiles. From the column of the most significant tile types, the tiles of type \( t_0^a \) transfer their eastern value to the western value until arriving to the \((n+k)\)-th column at the left. Note that, at the \((n+k-1)\)-th bit towards the left, a tile of type \( t_0^a = (d, 0, a, 0') \) or \( t_0^r = (d, 1, a, 1') \) (dark blue in Fig. 10) opens a new column that starts with \( t_0^a = (0', d, c, 0) \) or \( t_0^r = (1', d, c, 1) \) (light blue in Fig. 10) that transfers the value from the west to the north. Observe in Fig. 10 that these tiles form a light blue diagonal bisector between the column of the most significant bit below it, and the copied row tile number of \( N \). From the time that we pass these tiles, the values of each tile are copied to the north using \( t_1 \)-type tiles. Note that the tile types of the \((n+k)\)-th column are placed by matching with their southern and eastern tiles.

This process continues until the value of each tile bit is placed in a new tile bit in the same row as \( N \) but with a shift of \( n \) columns. In the terminal assembly, there is a copy of \( N \) (which was in positions \([(x, y), \ldots, (x+k, y)]\) in positions \([(x-k-1, y), \ldots, (x-1, y)]\).

### 3.4 The middle finding system

Let \( R \) be an explicitly bounded rectangle on \( \mathbb{Z}^2 \) i.e. a rectangle with horizontal sides that are bounded by specially marked tiles. The following lemma uses the IBC, DBC and U-turn systems to present a SFTAM TAS in order to find the middle of the height of \( R \). See Fig. 13.

**Lemma 15** (Middle finding system lemma). Let \( R \) be an explicitly bounded rectangle of width at least \( 3 \log(N) \), where \( N \) is the height of \( R \). There is a TAS \( S_{1/2} = (\Sigma, T_1/2, \sigma, \tau, \tau) \) such that for all assemblies with a seed located at the start, there is a tile \( t_m \) which appears only at coordinate \((x, \left\lfloor \frac{N}{2} \right\rfloor)\) with no other tiles to its left. Formally the system is:

- \( \Sigma = \Sigma_{IBC1} \cup \Sigma_{IBC2} \cup \Sigma_D \cup \Sigma_{DBC} \cup \{org, c\} \) where the labels in \( \Sigma \) are the same as in each system, except when a modification is explicitly mentioned. However, any two labels from two distinct systems are distinct from each other.
- \( T_U = T(IBC1) \cup T(IBC2) \cup T_U \cup T_{DBC} \cup \{t_u, t_c\} \)
Figure 13: The $k$-th stage of the assembly in the U-turn system is shown by yellow-filled rectangles. The value of the $k$-th significant bit is copied down by $k - 1$ rows during the previous stages. The $k$-th stage copies the value one time to the south and $n$ times to the left and finally $k$ times to the top. Here, $k = 5$ and $n = 8$. In addition, in the $k$-th stage, the tiles of type $t_{\downarrow}$ in the gray rectangle appear below the tile of type $t_{\uparrow}$, and they will be the supports for the $(k + 1)$-th stage. The seed is highlighted in the black rectangle.
The seed is $\sigma_+$ from the IBC System.

$\text{str} : \Sigma \cup \{\epsilon\} \rightarrow \{0,1,2\}$ such that $\text{str}(\epsilon) = 0$ and $\text{str}(\text{org}) = \text{str}(\epsilon) = 1$. The values of the $\text{str}$ function in $\Sigma$ are the same as each system.

$\tau = 2$

Proof. Let $R$ be an explicitly bounded rectangle and $N = 2^n + k$ with $k < 2^n$ be the height of $R$. Without loss of generality, we assume that the specially marked horizontal sides of $R$ being "Start" at the bottom and the other, "Finish", at the top, as in Fig. [14].

We use the IBC, DBC and U-turn systems from the previous sections to define our middle finding SFTM TAS that finds $N = 2^{(n-1)} + \frac{k}{2}$. There are two copies of the IBC, named $IBC_1$ and $IBC_2$. To avoid any confusion regarding tile types of IBC, DBC and U-turn systems each label is marked with its system name. For example, we use $1_{IBC}$ for the IBC system, and $1_{DBC}$ for the DBC system. However, whenever there is no ambiguity, we only use label 1 for simplicity. See an overview of the assembly in Fig. [14] to follow the structure of the proof. Now we describe the steps for the middle finding system.

0. Growing a column of tiles of type $t_{++}$ until "Finish".

The assembly starts to grow when $\sigma_+$ from the $IBC_1$ System is placed besides the starting tiles. The tiles of type $t_{++}$ pave the way until becoming blocked by "Finish" tiles, creating a column of tiles. These tiles are the support for the $IBC_1$ system.

1. Using the IBC System $IBC_1$ to find the height $N = 2^n + k$ of $R$.

Recall from Definition [13] of IBC systems, that each row tile number represents the height of the assembly along the $y$ axis. At the end, the assembly is blocked at the endpoint tiles of $R$ in the row tile number $N$.

We hereby modify the labels of the most significant bit tiles. We set $t_S = (1_S,0,1,1)$ instead of $t_S = (1_S,0,1,\epsilon)$, i.e the last label is changed to 1. This change will be used for starting the growth of the second IBC system. In addition, instead of $t_O = (1_s,\text{over},\epsilon,\epsilon)$ we have $t_O = (1_s,\text{over},\text{org},\text{org})$ ("org" stands for "orange"). The reason for this modification is that it will be useful in the 3rd step where we want to copy the value of $k$.

Here by this new modifications, immediately two types of tiles will grow. These tiles will be used later. Firstly, each time that in the IBC system a new most significant bit tile is added, a vertical tile bond grows in its south by $t_v = (\text{org},1,\text{org},1)$, where org is the new southern label of $t_O$, and the label 1 is $1_{\text{copy}}$. (They will be used in the copy system.) Secondly, when $t_v$ tiles collide with previous $t_v$ tiles, a double column of tiles of type $t_b$ appears. (They will be used as block tiles in the U-turn system.)

2. Returning to row number $2^n$ using the second IBC system $IBC_2$, which then outputs the value of $k$.

Afterwards, at the collision of $IBC_1$ with the finish block tiles, $IBC_2$ appears next to the most significant bit tile of $IBC_1$. This is where our modification of $IBC_1$ becomes useful. In fact, changing the fourth label of the most significant bit tile of the $IBC_1$ system to 1, enables this new IBC system $IBC_2$ to start growing, using the most significant column of $IBC_1$ as its support.

Note that due to the SFTM rotating tiles, the two IBC systems are able to grow in opposite directions. Thus, the assembly returns towards the row tile number of $2^n$ using the $IBC_2$ system.

Consequently, the value of the last row tile number of $IBC_2$ is $k$.

3. Copying $k$ until $2^{n-1}$ (the middle of $2^n$).

Now, the topmost column of tiles of type $t_\epsilon$ from the first step will act as the support for the third step. Thanks to the vertical tile bond that is located below the most significant bit tile of the $2^n$ row tile number, the $k$-row tile number is copied $2^{n-1}$ times until the $2^{n-1}$-row tile number of $IBC_1$ (this is half of $2^n$). To this aim, we use tiles of types $t_{\text{copy}} = (x,\text{copy},x,\text{copy})$ and $t_{\text{copy}}^* = (x,\text{copy},x,1_{DBC})$ (for $x \in \{0,1\}$) such that $x$ is the label of the IBC system, and $t_{\text{copy}}^*$ is used for copying the most significant bit tiles of the IBC system.

Since the column of tiles of type $t_\epsilon$ is the support of this step, the copy process stops after $2^{n-1}$ copies.

4. Halving $k$ by eliminating its least significant bit.

Now, the copied row tile number with value $k$ is halved by getting rid of its least significant bit. The block tiles that already appeared when two vertical tile bonds of type $t_\epsilon$ met each other, discard the least significant bit tile of the $k$-row tile number and copy the $\frac{k}{2}$-row tile number to the left of the first block tiles.
tiles. The tiles here are a tile of type $t_{u'} = (\text{org}, c, \epsilon, u)$ such that $u$ is a label with strength 2, and a seed $\sigma_u$ of the U-turn system that binds to it. These are block tiles. Also, from the south of $\sigma_u$, there are tiles of type $t_u$, the support tiles of the U-turn system. They are shown in black in Fig. 14.

5. **Shifting $\frac{k}{2}$ to the left by a U-turn system.**

The column of tiles of type $t_u$ from the previous step form the support for this step. The binding of the block tiles to the $\frac{k}{2}$-row tile number form a U-turn block seed and the $\frac{k}{2}$-row tile number shifts to its left. Remember that the U-turn system needs space for the $\frac{k}{2}$ tiles in the south of the U-turn seed; this condition is true at the row number $2^{n-1}$ along the y axis, since $\frac{k}{2} < 2^{n-1}$.

6. **Going up by $\frac{k}{2}$ using the DBC system.**

After shifting $\frac{k}{2}$ to the left at the row $\frac{2^n}{2}$, a DBC system starts from $\frac{k}{2}$ along the side of the Copy system. Note that here the label $1_{DBC}$ of $t_{\text{copy}}$ (from the Copy system) plays the role of the support tile type in this DBC system. Moreover, the first row of the DBC system i.e. the seed, is the copied $\frac{k}{2}$ row tile number from the U-turn system. We add two final modifications to the DBC. We change the tile type $t_{\text{copy}}$ to $t_m = (1', 0, 1', -)$ by changing the fourth label.

After passing the zero with the DBC system, a tile of type $t_m$ appears in row $\frac{N}{2} = 2^{n-1} + \frac{k}{2}$. The process of finding the middle finishes here by the appearance of $t_m$ in the assembly. Note that $t_m$ only appears once because the support of the DBC system is the left side of the copy system.

For an illustration, see Fig. 14 (where $t_m$ is in red).
4 Distinguishing order-1 cuboids by their genus using SFTAM

We now introduce a SFTAM tile assembly system $S_4$ which can distinguish the order-1 cuboids $C \in O_1$ (of genus 1) from the others in $O$. We now introduce a SFTAM seed is placed on a

Our main result is stated as follows.

**Theorem 16.** There is a SFTAM tile self-assembly system $S_4 = (\Sigma, T, \sigma, str, \tau)$ and a subset of tile-types $Y = \{t_{reg}, t_{mfs}\} \cup T_{ibc} \subseteq T$ such that for any order-1 cuboid $C = C_0 \ominus C_0'$ with the dimensions at least 10 for $C_0'$, if $S_4$ assembles on $C$ starting from a seed which is placed in a normal placement, the following holds:

- if $C$ has genus 0, then no tile of $Y$ appears in any producible assembly of $S$ on $C$, and
- if $C$ has genus 1, every terminal assembly of $S$ on $C$ contains at least one tile of $Y$.

There is a crucial point to consider: the system presented here works for the case where the assemblies’ seed is placed on a normal placement $p$, i.e. the place $p$ is “far enough” from the borders on the surface of the cuboid. Therefore first we define “normal placement”. Then, we explain a sketch of the construction of the assemblies before diving into the presentation of the system $S_4$.

**Definition 17** (Normal placement). Let $C$ be an order-1 cuboid such that $(x_1, y_2, z_2), (x_2, y_2, z_2), \ldots, (x_n, y_n, z_n)$ are its vertices. A placement $(x_p, y_p, z_p)$ is a normal placement of $C$ if and only if for all $i \in \mathbb{N}$, two of the following inequalities hold: $|x_i - x_p| \geq 3\log(N) + 6$, $|y_i - y_p| \geq 3\log(N) + 6$ and $|z_i - z_p| \geq 3\log(N) + 6$, where $N$ is the largest of the three dimensions of the cuboid. The set of all normal placements is denoted by $Pl_N(C)$.

The simplest example to demonstrate the concept of normal placement is on a cuboid $C \in O_0$. In this case, normal placements consist of the cuboid’s surface minus its “frame” i.e. the border of the cuboid’s edges with a thick margin. Hence there are 6 disconnected areas on $C$’s faces where the normal placements are. The normal placements on order-1 cuboids can be described similarly. It should be noted that in this case there are more than 6 disconnected areas. Recall the order-1 cuboids: the order-0 cuboid ($C \in O_0$), the order-1 cuboid with a pit ($C \in O_1^p$) or with a concavity ($C \in O_1^c$) whose genus are 0; and the order-1 cuboid with tunnel ($C \in O_1^t$) whose genus is 1. There are up to 9 disconnected areas with normal placements for $C \in C_0$, 10 for $C \in C_1$ and 11 for $C \in C_2$, depending on the new faces that are generated by the nature of the order-1 cuboid’s construction. We note that, by definition, when the smallest dimension is large enough with respect to the others, almost all the placements on order-1 cuboids are normal placements.

4.1 Description of the TAS $S_4$

We now describe $S_4$ formally.

**The tile types of $P_X$:**

- The seed $\sigma = (\epsilon, e, x', \epsilon)$
- $x' = (x', x'_e, x, x'_w)$ and $x = (x, x_e, x, x_w)$
- $x'_e = (x'_e, x'_e, x, x'_w)$ and $x'_w = (x'_w, x'_w, x_w, x'_w)$
- $x_e = (x_e, x_e, x_e, x_e)$ and $x_w = (x_w, x_w, x_w, x_w)$

![Figure 15: The tile types for the ribbon framing $P_X$.](image)

**The tile types for the ribbons framing $P_Y$:**
• The starter tile types: \( y_e = (x_e, y_e, x'_e, \epsilon) \) and \( y_u = (x_w, \epsilon, x'_w, y_w) \), \( y_e = (\epsilon, y_e, \epsilon, y_e) \) and \( y_w = (\epsilon, y_w, \epsilon, y_w) \).

• The tile types of four middle finding systems (presented in [13]) of \( \text{MFS}^u_\text{w}, \text{MFS}^d_\text{w} \) (east) with shared seed \( \sigma^+_e = (\text{over}, +, +, \text{over}, y_e) \) and support tiles of type \( t_{++} = (1, +, 1, +, +) \), and \( \text{MFS}^u_\text{w}, \text{MFS}^d_\text{w} \) (west) with shared seed \( \sigma^+_w = (\text{over}, y_w, \text{over}, +) \) and support \( t_{++} \). Apart from that, each system has its own tile types.

![Figure 16: The starter tile types, seeds and support tile type for the ribbon framing \( P_Y \).](image)

**The tile types of \( P_Z \):**

• \( t_{or} = (1, 1, 0', z''r) \), \( z_{1r} = (z_e, z''r, z''r, zr) \), \( z_{2r} = (z''r, \epsilon, zo, zo) \), \( z_e = (ze, zr, zr, zr) \), \( z_{or} = (r, zo, zo, zo) \)

• \( t_{ol} = (1, z''l, o', 1) \), \( z_{1l} = (z''l,zl,ze,z''l) \), \( z_{2l} = (zo, zo, zl', \epsilon) \), \( z_l = (l, zl, ze, zl) \), \( z_{ol} = (zo, zo, l, zo) \)

![Figure 17: The tile types for the ribbon framing \( P_Z \).](image)

**The tile types of inner filling:**

• \( t_{ee} = (ze, xe, ze, xe) \), \( t_{e1} = (ze, \epsilon, ze, xe) \), \( t_{ec2} = (ze, \epsilon, ze, xe) \), \( t_{ec3} = (ze, \epsilon, ze, xe) \), \( t_{ee4} = (ze, \epsilon, ze, xe) \)

• \( t_{eo} = (zo, xo, zo, xe) \), \( t_{eo1} = (zo, \epsilon, zo, xe) \), \( t_{eo2} = (zo, \epsilon, zo, xe) \)

• \( t_{we} = (ze, xe, ze, xe) \), \( t_{w1} = (ze, xe, ze, xe) \), \( t_{we2} = (ze, xe, ze, xe) \), \( t_{we3} = (ze, xe, ze, xe) \), \( t_{we4} = (ze, xe, ze, xe) \)

• \( t_{wo} = (zo, xo, zo, xo) \), \( t_{wo1} = (zo, xo, zo, xo) \), \( t_{wo2} = (zo, xo, zo, xo) \)

• \( t_{odd} = (zo, xo, zo, xo) \) and \( t_{even} = (ze, xe, ze, xe) \)

**The tile types which demonstrate a tunnel on the cuboid:**

• \( t_{reg} = (ze, xe, xo, zo, xe) \)

• \( T_{ibc} \) is not a single tile type, but it is a set of tile types: \( T_{ibc} = \{(i_1, i_2, i_3, i_4) \} \) where \( i_1, i_3 \in \{0, 1, \text{over}\} \), \( i_2 \) is a label from \( IBC1^u \cup IBC1^d \) and \( i_4 \) is a label from \( IBC1^u \cup IBC1^d \) systems of the first part of the middle finding systems.

• \( t_{mfs} = (ze, zo, ze, zo) \)
4.2 Overview of the assemblies of $S_G$

From now on, assume that the seed is placed on a normal placement of an order-1 cuboid $C$. An important point that impacts the growth of the assembly is the placement of the seed relative to the tunnel, if one exists. In general, an order-1 cuboid $C$ has 6 exterior faces. In an attempt to detect a potential tunnel whose entrances are on parallel faces, we need to separate them. For this purpose we use three planes $P_X$, $P_Y$, $P_Z$, one for each pair of parallel faces of $C$: $P_X$ is parallel to the plane formed by the $Y$-axis and $Z$-axis, $P_Y$ is parallel to the plane formed by the $X$-axis and $Z$-axis, and $P_Z$ is parallel to the plane formed by the $X$-axis and $Y$-axis. See Fig. 18 for an illustration.

![Figure 18: The ribbons framing planes $P_X$, $P_Y$ and $P_Z$. On the left, the traces of the planes. On the right, the shape of the ribbons' assembly.](image)

These planes themselves are obtained from being framed by several ribbons of tiles during the assembly. Based on this, we present the assemblies' growth process, consisting of an assembly's skeleton phase and a phase for filling up the skeleton.

1. Constructing the skeleton of the assembly’s structures by at most 7 perpendicular ribbons on $C$: Here, a step starts only when the previous step is finished.
   - one ribbon for framing the first plane $P_X$
   - two ribbons for framing the second plane $P_Y$
   - zero or four ribbons constitute the frame of the third plane $P_Z$ (depending on the intersection of the two previous planes, details will be given later)

2. Filling the inside of the assembly’s skeleton by distinctive tiles.

These ribbons are placed at the intersection of the planes and the cuboid. The difference $C \setminus \{P_X, P_Y, P_Z\}$ partitions the cuboid $C$ into up to eight connected components, called regions. The last step is for partially filling the interior of the regions by their distinctive tiles in a way that no connected component has a neighbor with the same inner filling tile.

In order to simplify the explanation of the assembly process, we first investigate the first phase to understand how the skeleton of the $S_G$ assemblies grows, depending on the seed’s placement on the cuboid, and how the skeleton partitions it into distinct connected components. Next, we study the second phase and afterwards, we present the main result.

4.3 Skeleton and regions

In the TAS $S_G$, the seed acts like a compass for the assemblies. When the seed is located at an arbitrary placement, without loss of generality, we assume that the side on which the seed is located is the face
parallel to the $XY$-plane and intersects the $Z$ axis, and the north label of the seed’s tile points towards the $Y$ axis.

In order to have an overall shape of the production when there is no obstacle in the assembly’s path, first we study the skeleton of the assembly on an arbitrary order-0 cuboid and we show how this method partitions it into eight distinct regions. Then we extend this to order-1 cuboids of genus 0. After that, we study the case of order-1 cuboids and we discuss the regions’ partition, when there is a potential tunnel.

4.3.1 Genus 0

Let $C \in O_0$ be an order-0 cuboid and assume that the seed $\sigma$ is placed at a normal placement $p \in \mathcal{P}l(C)$.

The structure of the skeleton.

We first show that every terminal assembly of $S_G$ on $C$ includes a “3-step skeleton” that consists of 7 perpendicular ribbons on $C$ and partitions $C$ into 8 regions such that the parallel faces are always in distinct regions.

In the first step, the assembly’s tiles make a vertical segment ribbon of tiles around $C$ starting at the south of $\sigma$. The seed $\sigma = (\epsilon, \epsilon, x', \epsilon)$ has label $x'$ with strength 2 at the south, and its other labels are $\epsilon$ with strength 0. A tile of type $x' = (x', x''_x, x, x''_w)$ sits at $\sigma$’s south and the label $x$ gives rise to a vertical ribbon of tiles of type $x = (x, x_x, x, x_w)$ such that it grows around $C$ and returns to the seed from the north. Moreover, two tiles of types $x'_x = (x'_x, x'_x, x, x''_x)$ and $x'_w = (x'_w, x'_w, x, x''_w)$ come at the east and the west of $x'$. By supporting the tile of type $x$, ribbons of tiles of types $x_x$ and $x_w$ form respectively in the south of $x'_x$ and $x'_w$. Now, there is a ribbon made of three columns of tiles of types $x_x$, $x$, and $x_w$ that build the frame for the plane $P_X$ (see Fig. 19). The plane $P_X$ framed by this segment ribbon divides $C$ into two regions, the right side and the left side of $C$ with respect to $\sigma$. (We always assume that we view the cuboid from the point of view of the $Z$-axis, as in Fig. 18 and thus left, right, up, down, back, front refer to this point of view.)

Therefore, the ribbons of plane $P_Y$ start to form when the ribbon of $P_X$ rebounds to the north of the seed using the tiles of type $y_1$, and $y_w$. Two segment ribbons starting from both right and left sides of $\sigma$ develop perpendicular to $P_X$ by using two middle finding systems (Definition 15), one on each side. To elaborate, the tiles of type $y_1 = (x_x, y_1, x'_x, \epsilon)$ and $y_w = (x_w, \epsilon, x'_w, y_1)$ sit in the assembly respectively between tiles of type $x'_x$ and $x_x$ at the east of $P_X$, and between tiles of type $x'_w$ and $x_w$ at the west of $P_X$. The tiles of type $y_2 = (\epsilon, y_2, \epsilon, y_1)$ and $y_w = (\epsilon, y_w, \epsilon, y_w)$ bind to $y_1$ and $y_w$, respectively. The tile types $y_e$ and $y_w$ are starting points for the eastern and western two-side IBC systems (Definition 12), that is, the seeds $\sigma_{en} = (\overline{\text{over}}, ++, \overline{\text{over}}, y_1)$ and $\sigma_{en} = (\overline{\text{over}}, \overline{\text{over}}, ++, y_2)$ make four IBC systems $IBC1_{en}^0$, $IBC1_{en}^0$ (east), and $IBC1_{en}^{0+}$, $IBC1_{en}^{0+}$ (west) with the support (1, ++, ++, ++). See Fig. 20 for an illustration. Note that each system has its own distinguished tile types. After that, when the two-side $IBC1$ ribbons of the plane $P_Y$ meet the $P_X$ ribbon, the tiles of types $x_e$ and $x_w$ of $P_X$ act as finish block tiles for the middle finding system.

Then, the four tiles of types $t_{eu} = (1, 1, \epsilon, x_e, 1)$, $t_{ed} = (1, 1, \epsilon, x_e, 1)$, $t_{wu} = (1, 1, \epsilon, x_w, 1)$ and $t_{wd} = (1, 1, \epsilon, x_w, 1)$ respectively, appear as the row tile number of 1 of the second IBC systems $IBC2_{eu}^0$, $IBC2_{eu}^0$ (east), and $IBC2_{ed}^{0+}$, $IBC2_{ed}^{0+}$ (west) of the middle finding systems. Note that the tiles of the second IBC2 systems have different labels from the ones of the $IBC1$ system (except the exterior labels of the most significant bit tiles in the $IBC1$ systems). On the same underlying rectangle as the underlying rectangle of $P_Y$, the ribbons grow and they stop in the middle of $P_Y$’s ribbons.

Once the $P_Y$ ribbons form, they separate $C$ into an up side and a down side. Thus, $C$ is now partitioned into four separate regions due to the first and second step ribbons.

Next, by finding the middle of each of $P_Y$’s ribbons on the right and left faces, four new perpendicular ribbons are generated from the tiles of type $t_{ml} = (1, 1, 0', 0', 1)$ at the left-up and right-down sides, and $t_{mr} = (1, 1, 0', 0', 1)$ at the right-up an left-down sides. They are the replacements of the tile of type $t_m$ of the middle finding systems. See Fig. 21 for an illustration. They go on, until they reach $P_X$ on the upper and down faces (see Fig. 22). The union of these four ribbon forms a frame for the plane $P_Z$. This step creates a separation between the front side and the back side of the cuboid $C$ with respect to $\sigma$.

Note that all the ribbons are able to form completely because by our hypothesis, the seed is located on a normal placement. This is an essential condition since we need enough space for using counters in the assembly. We show the assembly of $P_X$, $P_Y$ and $P_Z$ in Fig. 23.
Altogether, these three steps partition $C$ into 8 distinct areas which are noted by $R_{XYZ}$ such that $X, Y, Z \in \{0, 1\}$ where 0 represents the left, down and back sides, and 1 represents the right, up and front sides. For example, $R_{101}$ refers to the region at the right, down and front side of $C$, with respect to the seed. This way of partitioning $C$ helps to define the graph $G_C$, the region graph of $C$:

**Definition 18** (Region graph). Let $C$ be an order-1 cuboid. If an assembly of the TAS $S_G$ forms on $C$, there is a graph assigned to $C$ called the region graph $G_C$ whose vertices are the regions formed by the skeleton, and an edge is added between two regions if and only if they share a face of $C$.

The region graph for an order-0 cuboid is presented in Fig. 24. We observe that there is an edge in the graph $G_C$ if and only if a tunnel possibly exists between two regions, after subtracting an order-0 cuboid from $C$.

*Inner filling of the skeleton.*

After the formation of the skeleton, the fourth step is to fill the eight regions by lines of interior tiles, once the $P_Z$ ribbons reach $P_X$. For an order-0 cuboid $C$, $G_C$ is a bipartite graph that is 2-colorable. Thus, we use two tile types to distinctly tile $C$. First, four ribbons of tiles types appear at the intersection of $P_X$ and the $P_Z$ ribbons. These ribbons form by tile types of Fig 25.
4 Distinguishing Order-1 Cuboids by Their Genus Using Sftam

Figure 20: The formation of $P_Y$ (horizontal), out of $P_X$ (vertical).

Figure 21: Two of the ribbons framing $P_Z$.

From the parts along $P_X$, straight lines of tiles start growing parallel to the $x$ axis using strength 2 glues $x_{ev}$ and $x_{od}$. Thanks to modulo 5 (resp. 3) counters on the even (resp. odd) $P_X$ border tiles, there is one such line every other 5 (resp. 3) position along that part of the border with tiles of type $t_{ev} = (z_e, x_{ev}, z_e, x_{ev})$ (resp. $t_{od} = (z_o, x_{od}, z_o, x_{od})$). These lines form the even (resp. odd) filling tiles and fill the partitioned regions. See Fig. 26 for the tile types of partially inner filling of the regions and Fig. 27 for an illustration of the assembly.

The 2-coloring indicates also where the tiles of type $t_{even}$ and of type $t_{odd}$ can be placed. The region $R_{000}$ and the regions with even distance to it are tiled by tiles of type $t_{even}$, and the ones with odd distance to it, by $t_{odd}$ tiles. Therefore, the regions $R_{XYZ}$ ($X, Y, Z \in \{0,1\}$) are tiled with $t_{even}$ if and
only if the sum $X + Y + Z$ is even, and the regions with even sum are covered by $t_{\text{odd}}$. For more clarity, see Fig. 24 where the regions corresponding to $t_{\text{even}}$ are colored white and the ones with $t_{\text{odd}}$ are colored black in the region graph $G_C$ of the order-0 cuboid $C$.

4.3.2 Genus 0, with pit or concavity

Now, assume that we are on an order-1 cuboid $C \in O_1$ such that $C \in O_{1c}$ (the order-1 cuboids with concavity whose genus is 0), or $C \in O_{1p}$ (the order-1 cuboids with a pit whose genus is 0). In these cases, the assembly’s process is similar to the assembly on order-0 cuboids. The frame ribbons form completely by the assumption that the seed is located on a normal placement of $C$, the assembly’s skeleton is formed completely and separates $C$ into 8 distinct regions, and the insides of the regions are tiled independently by inner filling lines of tiles of types $t_{\text{odd}}$ and $t_{\text{even}}$. However, in the case of $O_{1c}$, the regions do not necessarily meet edge to edge, see Fig. 28 for an illustration.

4.3.3 Genus 1

We now explain the process in details for order-1 cuboids with a tunnel.

When the genus of $C$ is 1 (i.e. $C \in O_1$), the assembly has a skeleton too. The shape of the skeleton of the assembly on $C$ can be different depending on the placement of the seed regarding the tunnel. In the case where a tunnel intersects with some of the three planes, the number of disconnected regions can be less than 8. Consequently, the above defined graph $G_C$ of the cuboid is not bipartite when the genus is 1. We need some specific tiles that can appear in the assembly if and only if there is a tunnel on the cuboid. This is the key element of our proof.

Moreover, note that the three planes can intersect the tunnel in two ways: along the width or along the length of the tunnel. If a plane intersects along the length of the tunnel, its ribbon frames enter and pass inside the tunnel. If a plane intersects along the width of the tunnel, the plane acts like a separator between the two parallel faces where the tunnel’s entrances are located. Thus, a tunnel may have an intersection with up to two perpendicular planes, one along the length, and the other along the width.

Based on this, when there is a tunnel on the cuboid, three types of partitions into regions are possible, and the possible numbers of regions are 7, 5 and 1, respectively.

We study them case by case.

Case 1: One plane intersects along the width of the tunnel.

At least one of the planes $P_X$, $P_Y$ and $P_Z$ intersects with the tunnel of $C$, since its entrances are on parallel faces of the cuboid, and these planes are located between parallel faces. When the tunnel has
4 DISTINGUISHING ORDER-1 CUBOIDS BY THEIR GENUS USING SFTAM

Figure 23: The assembly of $P_X$, $P_Y$ and $P_Z$. The seed is located in the middle of the left part of $P_X$. $P_X$ grows fully before $P_Y$ starts growing. $P_Z$ assembles last.

an intersection with only one of the three planes, the plane intersects along the width of the tunnel. For example, in Fig. 29, the tunnel has an intersection with the plane $P_X$ only.

Lemma 19. Let $C = C_0 \setminus C'_0 \in O_1$ be an order-1 cuboid with the dimensions at least 10 for $C'_0$ and assume that the seed $\sigma$ is placed in a normal placement $p \in PL(C)$. In a terminal assembly of the system $S_G$, if exactly one plane has intersection with the tunnel, there exist 7 regions on the cuboid and a tile of type $t_{reg}$ or $t_{mfs}$ appears in the assembly.

Proof. Let the seed be placed in a manner that only one of the planes $P_X$, $P_Y$ or $P_Z$ intersects along the
Figure 24: The region graph $G_C$: $V(G_{S_G})$ are the distinct regions, and $E(G_{S_G})$ contains the edges between neighboring regions.

Figure 25: The tile types of four inner ribbons at the intersection of $P_X$ and the $P_Z$. width of the tunnel.
Moreover, if one of them runs head-first into an opposite line, the other cannot, because their distance must be at most 10. In particular, because any tile type \(t \notin \{C, \emptyset\}\) cannot see both an even tile and an odd tile through strength 1 sides. Then, this space can be filled by \(t_{\text{odd}}\) and \(t_{mfs}\) are the only tile type of \(S_G\) with labels \(z_0\) and \(z_o\) of inner filling tiles \(t_{\text{odd}}\) and \(t_{\text{even}}\). In Fig. 32, the places on \(C\) that reveal a tunnel by \(t_{\text{reg}}\) or \(t_{mfs}\) are shown. To conclude the proof, one needs to show that in a region with a disconnected border, there is a good empty space, that is an empty space which sees both an even tile and an odd tile through strength 1 sides. Then, this space can be filled by neither type of filling tiles, but it must eventually be filled by a tile of type \(t_{\text{reg}}\) or \(t_{mfs}\) appears. In a region with a tunnel, on each side of the tunnel, the border of every 10 \(\times\) 10 square must be crossed by either

- at least two of the lines of tiles starting from \(P_X\) on that side of the tunnel, or
- at least two of the lines exiting the tunnel.

In particular, because \(C_0^t\) is at least 10 \(\times\) 10 units wide, there are at least two lines crossing one of the edges the tunnel in the same direction. Each such line must either reach the opposite connected component of the border, be stopped orthogonally by a line from the opposite side of the tunnel, or run head-first into an opposite line. Consider such a pair of lines, with minimal distance between them. In particular, that distance must be at most 10.

- If one of the lines reaches the opposite connected component of the border, either of the spaces next to its end is good and in this case the tile of type \(t_{\text{reg}}\) appears in the assembly;
- likewise, if one of them is stopped orthogonally by a line from the opposite side of the tunnel, one of the spaces next to the intersections is good and a tile of type \(t_{mfs}\) appears.

Moreover, if one of them runs head-first into an opposite line, the other cannot, because their distance cannot be at the same time divisible by 15, positive and less than 10. Hence the pair satisfies one of the previous cases. This concludes the proof of that case of our construction.

**Case 2: the tunnel intersects with \(P_Z\), and exactly one of \(P_X\) and \(P_Y\).**

**Lemma 20.** Let \(C \in O_1^t\) be an order-1 cuboid and assume that the seed \(\sigma\) is placed in a normal placement \(p \in Pl(C)\). In a terminal assembly of the system \(S_G\), if \(P_Z\) and exactly one of \(P_X\) and \(P_Y\) have an intersection with the tunnel, there exist 5 regions on the cuboid and a tile of type \(t_{mfs}\) appears in the assembly.

**Proof.** If the seed is placed where the tunnel has intersection with two perpendicular planes, one of them intersects the tunnel along its width and the other one along its length. If \(P_Z\) intersects with the tunnel along the length, two ribbons of \(P_Z\) meet each other inside the tunnel. However, if \(P_Z\) intersects the tunnel along its width, they meet outside the tunnel.

In both cases, the tile \(t_{mfs} = (z_e, z_o, z_c, z_0)\) appears in the assembly when two frame ribbons of \(P_Z\) meet each other. Note that when the tunnel has intersection with \(P_Z\) and one of the planes \(P_X\) or \(P_Y\), the other plane separates the cuboid into two connected components such that one of them is a cuboid with genus 0 and the other one is a cuboid with genus 1. The part with genus 0 has 4 distinct regions, and the part with genus 1 (containing a tile of type \(t_{mfs}\)) has one single region. In fact, in this case, all four regions where these two planes are placed on unite. In total, there exist 5 distinct regions on the cuboid \(C\). For an illustration of the skeleton and its graph in this case, see Fig. 31.

Figure 26: The tile type \(t_{mfs}\) and \(t_{\text{reg}}\) (which may only appear if \(C\) has genus 1).
Case 3: The tunnel intersects with $P_X$ and $P_Y$.

**Lemma 21.** Let $C \in O_1^1$ be an order-1 cuboid and assume that the seed $\sigma$ is placed in a normal placement $p \in P(\mathcal{C})$. In a terminal assembly of the system $\mathcal{S}_{\mathcal{G}}$, if the planes $P_X$ and $P_Y$ intersect the tunnel, there exists 1 region on the cuboid and a tile of one of the $T_{ibc}$ types appears in the assembly.

**Proof.** In this case, the skeleton of the assembly is not the same as before. Recall the process of the assembly’s skeleton: The frame ribbons of the plane $P_X$ are generated independently from $\sigma$. Two segment ribbons of the plane $P_Y$ begin to grow after rebounding on the plane $P_X$, regardless of passing through a tunnel or not. However, the ribs of $P_Z$ start to grow only after finding the middle of $P_Y$ and they end by reaching the ribbons of $P_X$. Considering this process, when the two planes $P_X$ and $P_Y$ intersect with the tunnel, the plane $P_Z$ is not able to form since there is a tunnel that does not permit to have the collision of $P_Y$ and $P_X$, and the ribs of $P_Y$ do not end in $P_X$. As a result, since the plane $P_Z$ depends on the collision of the $P_Y$ ribs with the $P_X$ ribs, $P_Z$ is not able to form.

Moreover, two ribbons of $P_X$ must meet each other at a tile of one of the $T_{ibc}$ types that comes between their $IBC1$ systems. This happens inside the tunnel if $P_Y$ intersects the tunnel along its length, and outside the tunnel if it intersects the tunnel along its width. In either cases a tile of one of the $T_{ibc}$ types appears. See Fig. 34 for an illustration of the places where the presence of a tunnel entrance implies that we have a tile of one of the $T_{ibc}$ types.

Note that the skeleton consists of two closed loops of $P_X$ ribbons and $P_Y$ ribbons. This phenomenon demonstrates that the genus of $C$ is 1. In order to have a better overview, see Fig. 30. Furthermore, all regions are united and there is only one single region throughout the whole surface of $C$. \(\Box\)

Note that the situation when the seed is located inside the tunnel is similar to Case 3, up to topological isomorphism.

### 4.4 End of the proof

We are now ready to prove Theorem 16.

**Proof of Theorem 16.** Let $C = C_0 \setminus C_0^t \in O_1^1$ be an order-1 cuboid with the dimensions at least 10 for $C_0$ and $\alpha$ be an assembly of the TAS $\mathcal{S}_{\mathcal{G}} = (\Sigma, T, \sigma, \text{str}, \tau)$ such that its seed is placed at a normal placement. Note that if $C_0$ is too small there is no normal placement. We show that there is a tile type from $Y = \{t_{reg}\} \cup \{t_{mfs}\} \cup T_{ibc} \subseteq T$ in all terminal assemblies of $\mathcal{S}_{\mathcal{G}}$ on $C$ if and only if there is a tunnel on $C$ (i.e. its genus is 1).

If there is a tunnel on $C$, at least one of the planes $P_X$, $P_Y$, and $P_Z$ intersects with the tunnel since its entrances are on parallel faces of the cuboid, and these planes are located between parallel faces. If the tunnel of $C$ intersects with only one of the planes, due to Lemma 19, a tile of type $t_{reg}$ or $t_{mfs}$, which are the only tile types of $\mathcal{S}_{\mathcal{G}}$ with labels in common with both inner filling tile types $t_{odd}$ and $t_{even}$, appears in the assembly. In Fig. 32 the places where the presence of these tile types displays the presence of the tunnel is shown.

If two planes (among them $P_Z$) intersect with the tunnel on $C$, a tile of type $t_{mfs}$ appears in all terminal assemblies on $C$ by Lemma 20. See Fig. 33 for the places where the presence of a tile of type $t_{mfs}$ displays the presence of the tunnel.

If two planes $P_X$ and $P_Y$ intersect with the tunnel, Lemma 21 implies that a tile of one of the $T_{ibc}$ types is present in the assembly. See Fig. 34 for the places where the presence of a tunnel implies the presence of a tile of one of the $T_{ibc}$ types.

Overall, the places where a tunnel implies the presence of a tile of $Y$ are shown in Fig. 35.

Now, assume that one tile whose type is in $Y = \{t_{reg}\} \cup \{t_{mfs}\} \cup T_{ibc} \subseteq T$ exists in a terminal assembly of $\mathcal{S}_{\mathcal{G}}$ on $C$. We will show that there is a tunnel on the cuboid $C$.

Recall that, by Section 4.3.1, in a terminal assembly of $\mathcal{S}_{\mathcal{G}}$ on an order-1 cuboid with genus 0 partitions the cuboid into 8 distinct regions.

If a tile of type $t_{reg}$ is in the terminal assembly, its common labels with both inner filling tile types $t_{odd}$ and $t_{even}$ show that two regions are connected i.e. there are at most 7 distinct regions on $C$. Therefore, $C$ cannot have genus 0 (in fact, there is a tunnel between these two regions).

If a tile of type $t_{mfs}$ exists in a terminal assembly on $C$, there are two cases. In one case there is a tunnel that intersect only $P_X$ with width and $t_{even}$ and $t_{odd}$ intersect perpendicularly each other and as a
result \( t_{mfs} \) appears in the assembly. In the other case, two segment ribbons of \( P_Z \) must meet each other since the tiles whose labels correspond to the labels of \( t_{mfs} \) are those of the \( P_Z \) ribbons. Recall that the \( P_X \) and \( P_Y \) ribbons intersect at two places: one at the seed (since \( P_Y \) grows out of \( P_X \)) and a second time, where the tiles of type \( t_{eu}, t_{ed}, t_{wu} \) or \( t_{wd} \) appear in the assembly as the row tile number 1, in the second IBC system of the middle finding systems. The two ribbons of \( P_Z \), together with the parts of the middle finding system located between the second intersection of \( P_X \) and \( P_Y \) on the one hand, and \( P_Z \) on the other hand, form a closed ribbon on the surface of \( C \) (highlighted in green and blue Fig. 36). This ribbon and \( P_X \) pass through each other perpendicularly at only one place. Since they pass through each other perpendicularly, it can be concluded that the cuboid \( C \) cannot be topologically homeomorphic to the sphere, or in other words, be a genus 0 cuboid and a tunnel must exist.

Assume that a tile of one of the \( T_{ibc} \) types appears in the assembly. Note that two labels \( i_2 \) and \( i_4 \) of every tiles of one of the \( T_{ibc} \) types are the same as those of the tiles of the IBC1 systems in \( P_Y \) ribbons that are located opposite of each other (and no other tiles). Thus, the ribbons of \( P_Y \) must collide, so that there is a tile of one of the \( T_{ibc} \) types in the assembly and they do not reach the plane \( P_X \). Therefore, the frame ribbons \( P_X \) and \( P_Y \) have no intersection except at the seed. Since they pass through each other perpendicularly, as in the previous case, \( C \) cannot be topologically homeomorphic to a genus 0 cuboid. Therefore, a tunnel must exist so that \( P_X \) and \( P_Y \) do not intersect in two places.

5 Conclusion

We have introduced our new model, SFTAM, to perform tile self-assembly on 3D surfaces. We have showed that we can use self-assembly to determine the genus of a given surface. For this, we have worked on a simple and special family \( O_1 \) of polycubes, the order-1 cuboids.

It would be interesting to extend our results to a larger family of polycubes. For example, we can define \( O_n \) as the set of order-\( n \) cuboids for an integer \( n \), by a recursive definition: an order-\( n \) cuboid is obtained from the difference of an order-(\( n - 1 \))-cuboid and an order-0 cuboid. The genus of each order-\( n \) cuboid is at most \( n \). We think that we can extend our main theorem by designing SFTAM self-assembly systems that can distinguish the genus of order-\( n \) cuboids.

As future work, it would be interesting to study SFTAM tile self-assembly on other types of 3D surfaces, and characterize them by it.

References

[1] Z. Abel, N. Benbernou, M. Damian, E. D. Demaine, M. L. Demaine, R. Flatland, S. Kominers, and Robert Schweller. Shape replication through self-assembly and RNase enzymes. Proceedings of the 21st Annual ACM- SIAM Symposium on Discrete Algorithms (SODA 2010), pages 1045–1064, Austin, Texas, January 17–19 2010.

[2] G. Aloupis, P. Bose, S. Collette, E. D. Demaine, M. L. Demaine, L. Doušeb, V. Dujmović, J. Iacono, S. Langerman and P. Morin. Common unfoldings of polyominoes and polycubes. Proceedings of the International Conference on Computational Geometry, Graphs and Applications, CGGA’10. Lecture Notes in Computer Science 7033:44–54, 2010.

[3] R. D. Barish, R. Schulman, P. W. K. Rothemund and E. Winfree. An information-bearing seed for nucleating algorithmic self-assembly. Proceedings of the National Academy of Sciences 106(15):6054–6059, 2009.

[4] M. Cook, T. Stéрин and D. Woods. Small tile sets that compute while solving mazes. Proceedings of the 27th International Conference on DNA Computing and Molecular Programming (DNA 27), LIPIcs 205, 8:1–8:20, 2021.

[5] J. Durand-Lose, J. Hendrickx, M. J. Patitz, I. Perkins and M. Sharp. Self-assembly of 3-D structures using 2-D folding tiles. Natural Computing 19, 337–355, 2020.

[6] M. B. Jones, N. C. Seeman and C. A. Mirkin. Programmable Materials and the Nature of the DNA Bond. Science 347:840–840, 2015.
[7] M. Y. Kao and V. Ramachandran. DNA self-assembly for constructing 3D boxes. Proceedings of the 12th International Symposium on Algorithms and Computation, ISAAC’01, Lecture Notes in Computer Science 2223:429–441, 2001.

[8] W. Liu, H., Zhong, R., Wang and N. C. Seeman. Crystalline two-dimensional DNA-origami arrays. Angewandte Chemie International Edition 50(1):264–267, 2011.

[9] M. J. Patitz. An introduction to tile-based self-assembly and a survey of recent results. Natural Computing 13, 195–224, 2014.

[10] P. W. K. Rothemund. Folding DNA to create nanoscale shapes and patterns. Nature 440(7082):297–302, 2006.

[11] P. W. K. Rothemund and E. Winfree. The Program-Size Complexity of Self-Assembled Squares. Proceedings of the Thirty-Second Annual ACM Symposium on Theory of Computing, STOC 2000, 459–468, 2000.

[12] V. A. Sontakke and Y. Yokobayashi. Programmable Macroscopic Self-Assembly of DNA-Decorated Hydrogels. Journal of the American Chemical Society 144 (5), 2149-2155, 2022.

[13] DNA Nanotechnology: Bibliography from Ned Seeman’s Laboratory. http://seemanlab4.chem.nyu.edu/nanobib.html

[14] H. Wang. Proving theorems by pattern recognition – II. Bell System Technical Journal 40 (1):1–41, 1961.

[15] E. Winfree. Algorithmic Self-Assembly of DNA. Ph.D.thesis, California Institute of Technology, 1998.

[16] J.F. Woods, L. Gallego, P. Pfister, M. Maaloum, A.V. Jentzsch, and M. Rickhaus. Shape-assisted self-assembly. Nat. Commun. 13, 368 , 2022.

[17] S.Zhung, Fabrication of novel biomaterials through molecular self-assembly. Nat Biotechnol 21, 1171–1178 , 2003.

[18] R. Zhuo, F. Zhou, X. He, R. Sha, N. C. Seeman and P. M. Chaikin. Litters of self-replicating origami cross-tiles. Biophysics and Computational Biology 116 (6) 1952–1957, 2019.
Figure 27: The inner filling with tiles of types $t_{\text{even}}$ (white) and $t_{\text{odd}}$ (black) at the two places where the $P_Z$ ribbons meet $P_X$. 
Figure 28: Cuboid with concavity: The three planes $P_X$ (red), $P_Y$ (green) and $P_Z$ (blue), which consists of two semi-planes.

(a) The region graph

(b) The cuboid and the three planes.

Figure 29: The case where $C \in O_1$ is partitioned into 7 distinct regions. If there is a tunnel between two distinct regions, a tile of type $t_{reg}$ or $t_{mfs}$, which have common labels with both $t_{even}$ and $t_{odd}$, must appear in the assembly.

(a) The tunnel intersects along the length of plane $P_X$ and width of plane $P_Y$.

(b) The tunnel intersects along the width of plane $P_X$ and length of plane $P_Y$.

Figure 30: Intersection of tunnel with two planes $P_X$ (red) and $P_Y$ (green).
(a) The region graph

(b) The tunnel intersects along the width of plane $P_X$ and length of plane $P_Z$.

(c) The tunnel intersects along the length of plane $P_X$ and width of plane $P_Z$.

(d) A tile of type $t_{mfs}$ appears if and only if two segments of $P_Z$ (in purple) intersect each other by passing through a tunnel (instead of reaching $P_X$).

Figure 31: The case where $C \in O^t$ and $C$ is partitioned into 5 distinct regions.
Figure 32: The places on an order-1 cuboid that, if a tunnel is placed there, a tile of type $t_{reg}$ on $t_{mfs}$ appears.

Figure 33: The places on an order-1 cuboid where, if a tunnel is placed there, a tile of type $t_{mfs}$ appears.
Figure 34: The places on $C$ where $t_{ibc}$ displays the presence of a tunnel on $C$. Note that the tunnel appears by $t_{ibc}$ also when the seed is inside the tunnel, since up to topological isomorphism, it is the same case.

Figure 35: The places on a cuboid where, if there is a tunnel, a tile of $Y$ must appear in the assembly.
Figure 36: The closed ribbon formed by parts of the middle finding system (green) and the two ribbons of $P_Z$ (blue), when two $P_Z$ ribbons meet each other instead of reaching $P_X$. They meet the red ribbon $P_X$ only once.