Geometric tiles and powers and limitations of geometric hindrance in self-assembly

Daniel Hader · Matthew J. Patitz

Accepted: 27 January 2021 / Published online: 15 March 2021 © The Author(s), under exclusive licence to Springer Nature B.V. part of Springer Nature 2021

Abstract

Tile-based self-assembly systems are capable of universal computation and algorithmically directed growth. Systems capable of such behaviors typically make use of “glue cooperation” in which the glues on at least 2 sides of a tile must match and bind to those exposed on the perimeter of an assembly for that tile to attach. However, several models have been developed which utilize “weak cooperation”, where only a single glue needs to bind but other preventative forces (such as geometric, or steric, hindrance) provide additional selection for which tiles may attach, and where this allows for algorithmic behavior. In this paper we first work in a model where tiles are allowed to have geometric bumps and dents on their edges. We show how such “geometric” tiles can simulate systems of square tiles with complex glue functions (using asymptotically optimal sizes of bumps and dents). We then show that at scale factor 1 it is impossible for geometric tiles to simulate the behavior of systems which can include duples (i.e. tiles either twice as long or twice as tall as square tiles), and also that with only weak cooperation via geometric hindrance, no system in any model can simulate even a class of tightly constrained, deterministic cooperative systems. This helps to further define the limits of the powers of systems relying on geometric hindrance instead of glue cooperation. A shortened version of this paper appeared in the proceedings of the 18th International Conference on Unconventional Computation and Natural Computation (UCNC 2019).

1 Introduction

Systems of tile-based self-assembly in models such as the abstract Tile Assembly Model (aTAM) Winfree (1998) have been shown to be very powerful in the sense that they are computationally universal Winfree (1998) and are also able to algorithmically build complex structures very efficiently Rothemund and Winfree (2000), Soloveichik and Winfree (2007). The key to their computational and algorithmic power arises from the ability of tiles to convey information via the glues that they use to bind to growing assemblies and the preferential binding of some types of tiles over others based upon the requirement that they simultaneously match the glues of multiple tiles already in an assembly. This is called (glue) cooperation, and in physical implementations it can require a difficult balance of conditions to enforce.

It is conjectured that systems which do not utilize cooperation, that is, those in which a tile can bind to a growing assembly by matching only a single glue, do not have the power to perform computations or algorithmically guided growth Doty et al. (2011), Meunier and Woods (2017), Meunier et al. (2014). However, several past results have shown that a middle ground, which we call weak cooperation, can be used to design systems which are capable of at least some of the powers of cooperative systems. It has been shown that using geometric hindrance Fu et al. (2012), Hendricks et al. (2015), Gilbert et al. (2016), Fekete et al. (2015) or repulsive glue forces Reif et al. (2005), Patitz et al. (2011), Luchsinger et al. (2018), Hendricks et al. (2016), systems with a binding threshold of 1 (a.k.a. temperature-1 systems) are capable of universal computation. This is because they are able to simulate temperature-2 zig-zag aTAM systems, which are in many ways the most restrictive and deterministic of aTAM systems, but which are still capable of simulating arbitrary Turing machines.
In this paper, we further explore some of the powers and limitations of self-assembly in weakly-cooperative systems. First, we investigate the abilities of so-called geometric tiles (those with bumps and dents on their edges), which were shown in Fu et al. (2012) to be able to self-assemble \( n \times n \) squares using only temperature-1 and \( \Theta(\sqrt{\log n}) \) unique tile types (beating the lower bound of \( \log n / \log \log n \) required for square aTAM tiles), and also at temperature-1 to be able to simulate temperature-2 zig-zag systems in the aTAM, and thus arbitrary Turing machines. Here we prove their ability to simulate non-cooperative, temperature-1, aTAM systems that have flexible glue functions (those that allow glue types to bind to arbitrary sets of other glue types). We provide a construction and then show that it uses the minimum possible number of unique glue types (which is 2), and that it is asymptotically optimal with respect to the size of the geometries used. We also show that geometric tiles are capable of simulating aTAM systems with arbitrary flexible glue functions at any temperature.

The Dupled abstract Tile Assembly Model (DaTAM) is similar to the aTAM, but with the addition of duple tiles which are simply \( 2 \times 1 \), or \( 1 \times 2 \), rectangular tiles. While this addition may seem minimal, the DaTAM allows for weak cooperation via geometric hindrance, and is capable of universal computation at temperature-1 Hendricks et al. (2015). In a previous version of this paper Hader and Patitz (2019), we incorrectly claimed that every temperature-1 DaTAM system can be simulated by some temperature-1 GTAM system. Here, we instead prove that at scale factor 1 such simulation is in fact impossible.

Our final contribution is to expose a fundamental limitation of weakly cooperative self-assembling systems which rely on geometric hindrance. As previously mentioned, they are able to simulate the behaviors of temperature 2 aTAM systems which are called zig-zag systems. These systems have the properties that at all points during assembly, there is exactly one frontier location where the next tile can attach (or zero once assembly becomes terminal), and for every tile type, every tile of that type which attaches does so by using the exact same input sides (i.e. the initial sides with which it binds), and also has the same subset of sides used as output sides (i.e. sides to which later tiles attach).\(^1\) It has previously been shown in Hendricks et al. (2015) that there exist temperature-2 aTAM systems which cannot be simulated by temperature-1 systems with duples, but that proof fundamentally utilizes a nondeterministic, undirected aTAM system with an infinite number of unique terminal assemblies. Here we try to find a “tighter” gap and so we explore aTAM temperature-2 systems which are directed and only ever have a single frontier location and whose tile types always have fixed input sides, but we make the slight change of allowing for tiles of the same type to sometimes use different output sides. We prove that with this minimal addition of uncertainty, no system which utilizes weak cooperation with geometric hindrance can simulate such a system, at any scale factor. Thus, while geometric hindrance is an effective tool for allowing the simulation of general computation, the dynamics which such weakly-cooperative systems can capture is severely restricted.

### 2 Preliminaries

In this section we provide brief descriptions of the models used in this paper. References are provided for more thorough definitions.

#### 2.1 Informal description of the abstract Tile Assembly Model

This section gives a brief informal sketch of the abstract Tile Assembly Model (aTAM) Winfree (1998) and uses notation from Rothemund and Winfree (2000) and Lathrop et al. (2009). For more formal definitions and additional notation, see Rothemund and Winfree (2000) and Lathrop et al. (2009).

A **tile type** is a unit square with four sides, each consisting of a **glue label** which is often represented as a finite string. An aTAM system has a finite set \( T \) of tile types, but an infinite number of copies of each tile type, with each copy being referred to as a **tile**. A **glue function** is a symmetric mapping from pairs of glue labels to a non-negative integer value which represents the strength of binding between those glues. An **assembly** is a positioning of tiles on the integer lattice \( \mathbb{Z}^2 \), described formally as a partial function \( \alpha : \mathbb{Z}^2 \leftarrow T \). Let \( A_T^2 \) denote the set of all assemblies of tiles from \( T \), and let \( A_T^\infty \) denote the set of finite assemblies of tiles from \( T \). We write \( \alpha \subseteq \beta \) to denote that \( \alpha \) is a subassembly of \( \beta \), which means that dom \( \alpha \subseteq \text{dom } \beta \) and \( \alpha(p) = \beta(p) \) for all points \( p \in \text{dom } \alpha \). Two adjacent tiles in an assembly interact, or are

\(^1\) Note that **rectilinear** systems can also be simulated, and they have similar properties except that they may have multiple frontier locations available.
attached, if the glue labels on their abutting sides have positive strength between them according to the glue function. Each assembly induces a binding graph, a grid graph whose vertices are tiles, with an edge between two tiles if they interact. The assembly is $\tau$-stable if every cut of its binding graph has strength at least $\tau$, where the strength of a cut is the sum of all of the individual glue strengths in the cut.

A tile assembly system (TAS) is a 4-tuple $T = (T, \sigma, G, \tau)$, where $T$ is a finite set of tile types, $\sigma : \mathbb{Z}^2 \rightarrow T$ is a finite $\tau$-stable seed assembly, $G$ is a glue function, and $\tau$ is the temperature parameter (a.k.a. binding threshold). In the case that the glue function $G$ is diagonal, meaning that each glue only has a non-zero strength with itself, $G$ is often omitted from the definition and a TAS is defined as the triple $T = (T, \sigma, \tau)$ where the strengths between identical glues are given as part of $T$. Glue functions which are not diagonal are often said to define flexible glues. Given an assembly $\alpha$, the frontier, $\partial^F \alpha$, is the set of locations to which tiles can $\tau$-stably attach. An assembly $\alpha$ is producible if either $\alpha = \sigma$ or if $\beta$ is a producible assembly and $\alpha$ can be obtained from $\beta$ by the stable binding of a single tile to a location in $\partial^F \beta$. In this case we write $\beta \rightarrow^T \alpha$ (to mean $\alpha$ is producible from $\beta$ by the attachment of one tile), and we write $\beta \rightarrow^T \alpha$ if $\beta \rightarrow^T \alpha$ (to mean $\alpha$ is producible from $\beta$ by the attachment of zero or more tiles). An assembly sequence in a TAS $T$ is a (finite or infinite) sequence $\vec{\alpha} = (\alpha_0, \alpha_1, \ldots)$ of assemblies in which each $\alpha_{i+1}$ is obtained from $\alpha_i$ by the addition of one tile, i.e. $\alpha_i \rightarrow^T \alpha_{i+1}$. We let $A[T]$ denote the set of producible assemblies of $T$. An assembly $\alpha$ is terminal if no tile can be $\tau$-stably attached to it, i.e. $|\partial^F \alpha| = 0$. We let $A_\infty[T] \subseteq A[T]$ denote the set of producible, terminal assemblies of $T$. A TAS $T$ is directed if $|A_\infty[T]| = 1$. Hence, although a directed system may be nondeterministic in terms of the order of tile placements, it is deterministic in the sense that exactly one terminal assembly can be produced. We say that a system $T$ is a single-assembly-sequence system (SASS), if for every producible assembly $\alpha \in A[T], |\partial^F \alpha| \leq 1$, i.e. there is never more than one frontier location on any assembly producible from the seed. If a system $T$ is a SASS and also directed, then it is fully deterministic. We say that a system $T$ is a zig-zag system if it is a SASS where, for every producible assembly $\alpha \in A[T]$ and $\beta \subseteq \alpha$, the $y$ coordinate of $\partial^F \alpha$ is never smaller than the $y$ coordinate of $\partial^F \beta$.

### 2.2 Informal description of the Geometric Tile Assembly Model

The geometric tile assembly model (GTAM) is similar to the aTAM with the addition of geometric bumps along the sides of tiles and the restriction that the glue function be diagonal. This section will provide an informal introduction to the model, but a more complete introduction can be found in Fu et al. (2012). The introduction presented here differs slightly from that in Fu et al. (2012) in that we focus only on 1-dimensional geometry and try to match the notation as closely as possible to the aTAM definition in the previous section.

A geometry of size $n$ is a mapping from $\{1, \ldots, n\}$ to $\{0, 1\}$. This represents $n$ possible locations for bumps with a 1 representing a bump at that location and a 0 representing no bump. A geometric tile type is a unit square with a glue label and a geometry on each side. Note that in the standard definition of this model, a glue label can be thought of as covering the entire side of a tile (i.e. only a single glue label can occur on any given side). However, a more generalized version allows for multiple instances of glues on each side, and such a model is considered for Theorem 4 in Sect. 4. Similar to tiles in the aTAM, two geometric tiles interact if the glue labels on their abutting sides have a positive strength; however, if the two abutting geometries have bumps in corresponding locations, they are called incompatible and cannot bind regardless of glue strength. (See Fig. 2 for an example.) It's important to note that opposite sides of a tile can possess the same geometry. Geometry is rotated along the sides of the tile, so in this case, the geometry on the opposite side would be reversed. Thus any geometry which has a pair of...
bumps symmetric about its middle would be incompatible with itself. A geometric tile assembly system (GTAS) is a triple \( T = (T, \sigma, \tau) \), where \( T \) is a finite set of geometric tile types, \( \sigma : \mathbb{Z}^2 \rightarrow T \) is a finite \( \tau \)-stable seed assembly, and \( \tau \) is the temperature. Since the glue function for a GTAM system is diagonal, it is omitted from the definition for convenience. Also note that the sizes of all geometries in a given GTAS are the same.

### 2.3 Additional models

A wide variety of models which generalize and extend certain aspects of the aTAM have been developed. Of those, due to space constraints we will briefly mention a few and cite references which can be used to find full definitions.

The 3D aTAM Cook et al. (2011) is the natural extension of the aTAM from 2-dimensional square tiles to 3-dimensional cubes. In Hendricks et al. (2015), the Dupled abstract Tile Assembly Model (DaTAM) is defined as an extension to the aTAM which allows for not only the standard square, \( 1 \times 1 \), tiles, but also the inclusion of “duples” (or dominos) which are tiles of dimension \( 2 \times 1 \) or \( 1 \times 2 \). Allowing for more complex tile shapes, the Polyomino Tile Assembly Model (polyTAM) Fekete et al. (2015) allows for tiles composed of arbitrary numbers of unit squares which are connected along aligned faces. The polygonal TAM Gilbert et al. (2016), Demaine et al. (2014) allows for tiles to have arbitrary polygonal shapes, and this is the only model mentioned which doesn’t have an underlying regular lattice. Finally, another extension to the aTAM which we discuss is one which includes negative glues, or glues which exhibit repulsive forces Patitz et al. (2011). In systems with negative glues, two tiles may have adjacent faces with matching glues whose interaction strength is a negative integer. This is subtracted from the overall sum of binding strengths of adjacent glues to determine if the tile can attach.\(^2\)

### 2.4 Cooperation

Self-assembling systems in tile assembly models contain a parameter known as the minimum binding threshold, often called the temperature. This parameter specifies the minimum binding strength, summed over all binding glues, that a tile must have with an assembly in order to attach to it. The binding strengths of glues are typically discrete, positive integer values. If the temperature parameter is set to 2 or greater, we say that the system is strongly-cooperative, uses strong cooperation, or uses glue cooperation, because it is possible for the attachment of a new tile to an assembly to require that it bind its glues, with positive affinity, to more than one tile already existing in the assembly. For example, in a temperature-2 system (i.e. one where the temperature parameter \( \tau = 2 \)), a tile may attach by binding with two glues each of strength 1, and thus two tiles already in the assembly cooperate to allow for the new attachment.

In contrast, we say that a system is non-cooperative if its temperature parameter is set to 1 and in all situations where there is an empty location with an exposed incident glue, any tile with a matching glue can attach there. Finally, we say a system is weakly-cooperative, or uses weak-cooperation, if its temperature parameter is set to 1 but it is able to make use of any form of binding hindrance. In such a system, the binding strength of any single glue is strong enough to allow a tile to attach to an assembly; however, it is possible that some tiles are prevented from binding by another factor. Two such forms of binding hindrance we’ll consider are geometric hindrance and glue repulsion. In this context, geometric (a.k.a. steric) hindrance occurs when a tile cannot bind in a location because at least some of the space that it would occupy is already occupied by some portion of another tile. This is relevant when tiles have more complex shapes than unit squares. See Fig. 3 for an example, as well as Fu et al. (2012), Gilbert et al. (2016), Fekete et al. (2015), Hendricks et al. (2015). Glue repulsion occurs when glues are able to experience negative strength interactions, i.e. when they can repel each other and their interaction subtracts from the total binding strength of a tile to an assembly. Examples can be found in Patitz et al. (2011), Doty et al. (2013), Luchsinger et al. (2017). An example of the combination of geometric hindrance and glue repulsion can be found in Hendricks et al. (2016).

### 2.5 Simulation

First, we give a very brief intuitive definition of what it means for one tile assembly system to simulate another, and then in the next subsection provide more technically detailed defi-

---

\(^2\) Note that with negative glues, more complex dynamics, which include the breaking apart of assemblies, are possible Luchsinger et al. (2018).
nitions related to simulation, especially as it relates to scale factors greater than 1.

Intuitively, simulation of a system \( T \) by a system \( S \) requires that there is some scale factor \( m \in \mathbb{Z}^+ \) such that \( m \times m \) squares of tiles in \( S \) represent individual tiles in \( T \), and there is a “representation function” capable of inspecting assemblies in \( S \) and mapping them to assemblies in \( T \). A representation function \( R \) takes as input an assembly in \( S \) and returns an assembly in \( T \) to which it maps. In order for \( S \) to correctly simulate \( T \), it must be the case that for any producible assembly \( \alpha \in \mathcal{A}[T] \) that there is a corresponding assembly \( \beta \in \mathcal{A}[S] \) such that \( R(\beta) = \alpha \). (Note that there may be more than one such \( \beta \).) Furthermore, for any \( \alpha' \in \mathcal{A}[T] \) which can result from a tile addition to \( \alpha \), there exists \( \beta' \in \mathcal{A}[S] \), where \( R(\beta') = \alpha' \), which can result from the addition of one or more tiles to \( \beta \), and conversely, \( \beta \) can only grow into assemblies which can be mapped into valid assemblies of \( T \) into which \( \alpha \) can grow.

2.5.1 Technical definitions for simulation

This section contains a formal, rigorous definition of what it means for one tile assembly system to “simulate” another. Our definitions come from Meunier et al. (2014), but we make slight modifications to account for the simulation of geometric tiles. Also, note that a great amount of the complexity required for the definitions arises due to the possible dynamics of simulations with scale factors > 1, and that otherwise the mapping of assemblies and equivalence of production and dynamics are much more straightforward.

From this point on, let \( T \) be a tile set, and let \( m \in \mathbb{Z}^+ \). An \( m \)-block supertile over \( T \) is a partial function \( \alpha : \mathbb{Z}_m^2 \rightarrow T \), where \( \mathbb{Z}_m^2 = \{0, 1, \ldots, m-1\} \). Let \( B^T_m \) be the set of all \( m \)-block supertiles over \( T \). The \( m \)-block with no domain is said to be empty. Note that, as depicted in Fig. 5 and 4, geometric tiles consist of tile bodies and geometry regions. This allows the portions of each geometric tile to be (sub-)addressed within a \((2k + j) \times (2k + j)\) region. However, for simplicity we still consider geometric tiles as being in the plane \( \mathbb{Z}^2 \) with the center of each tile at a point in \( \mathbb{Z}^2 \) and the geometry regions of adjacent tiles overlapping. For a general assembly \( \alpha : \mathbb{Z}^2 \rightarrow T \) and \((x, y) \in \mathbb{Z}^2\), define \( \alpha^m_{x,y} \) to be the \( m \)-block supertile defined by \( \alpha^m_{x,y}(i_x, i_y) = \alpha(mx + i_x, my + i_y) \) for \( 0 \leq i_x, i_y < m \). For some tile set \( S \), a partial function \( R : B^S_m \rightarrow T \) is said to be a valid \( m \)-block supertile representation from \( S \) to \( T \) if for any \( \alpha, \beta \in B^S_m \) such that \( \alpha \subseteq \beta \) and \( \alpha \in \text{dom} \ R \), then \( R(\alpha) = R(\beta) \).

For a given valid \( m \)-block supertile representation function \( R \) from tile set \( S \) to tile set \( T \), define the assembly representation function \( R^* : \mathcal{A}^S \rightarrow \mathcal{A}^T \) such that \( R^*(\alpha') = \alpha \) if and only if \( \alpha(x, y) = R(\alpha^m_{x,y}) \) for all \((x, y) \in \mathbb{Z}^2\). For an assembly \( \alpha' \in \mathcal{A}^S \) such that \( R(\alpha') = \alpha \), \( \alpha' \) is said to map cleanly to \( \alpha \in \mathcal{A}^T \) under \( R^* \) if for all non empty blocks \( \alpha^m_{x,y} \), \((x, y) + (u_x, u_y) \in \text{dom} \ \alpha \) for some \( u_x, u_y \in \{-1, 0, 1\} \) such that \( u_x^2 + u_y^2 \leq 1 \), or if \( \alpha' \) has at most one non-empty \( m \)-block \( \alpha^m_{0,0} \). In other words, \( \alpha' \) may have tiles on supertile blocks representing empty space in \( \alpha \), but only if that position is adjacent to a tile in \( \alpha \). We call such growth “around the edges” of \( \alpha' \) fuzzy and thus restrict it to be adjacent to only valid supertiles, but not diagonally adjacent (i.e. we do not permit diagonal fuzzy).\(^4\)

In the following definitions, let \( T = (T, \sigma_T, \tau_T) \) be a tile assembly system, let \( S = (S, \sigma_S, \tau_S) \) be a tile assembly system, and let \( R \) be a valid \( m \)-block representation function \( R : B^S_m \rightarrow T \).

**Definition 1** We say that \( S \) and \( T \) have equivalent productions (under \( R \)), and we write \( S \leftrightarrow T \) if the following conditions hold:

1. \( \{R^*(\alpha') | \alpha' \in \mathcal{A}[S]\} = \mathcal{A}[T] \).
2. \( \{R^*(\alpha') | \alpha' \in \mathcal{A}^\square[S]\} = \mathcal{A}^\square[T] \).
3. For all \( \alpha' \in \mathcal{A}[S], \alpha' \) maps cleanly to \( R^*(\alpha') \).

**Definition 2** We say that \( T \) follows \( S \) (under \( R \)), and we write \( T \rightarrow_R S \) if \( \alpha' \rightarrow^S \beta' \) for some \( \alpha', \beta' \in \mathcal{A}[S] \), implies that \( R^*(\alpha') \rightarrow^T R^*(\beta') \).

\(^3\) Note that \( R^* \) is a total function since every assembly of \( S \) represents some assembly of \( T \); the functions \( R \) and \( \alpha \) are partial to allow undefined points to represent empty space.

\(^4\) With an exception which is noted in Sect. 5.
Definition 3 We say that $S$ models $T$ (under $R$), and we write $S \models_R T$, if for every $\alpha \in A[T]$, there exists $\Pi \subseteq A[S]$ where $\Pi \neq \emptyset$ and $R^*(\alpha') = \alpha$ for all $\alpha' \in \Pi$, such that, for every $\beta \in A[T]$ where $\alpha \rightarrow^T \beta$, (1) for every $\alpha' \in \Pi$ there exists $\beta' \in A[S]$ where $R^*(\beta') = \beta$ and $\alpha' \rightarrow^S \beta'$, and (2) for every $\alpha'' \in A[S]$ where $\alpha'' \rightarrow^S \beta'$, $\beta' \in A[S]$, $R^*(\alpha'') = \alpha$, and $R^*(\beta') = \beta$, there exists $\alpha' \in \Pi$ such that $\alpha' \rightarrow^S \alpha''$.

The previous definition essentially specifies that every time $S$ simulates an assembly $\alpha \in A[T]$, there must be at least one valid growth path in $S$ for each of the possible next steps that $T$ could make from $\alpha$ which results in an assembly in $S$ that maps to that next step.

Definition 4 We say that $S$ simulates $T$ (under $R$) if $S \not\models_R T$ (equivalent productions), $T \rightarrow^R S$ and $S \models_R T$ (equivalent dynamics).

3 Simulation of temperature-1 aTAM systems

Theorem 1 For any temperature-1 aTAM system, $T = (T, \sigma_T, G_T, 1)$, with arbitrary symmetric glue function $G_T$ (a.k.a. flexible glues), there exists a temperature-1 GTAM system $U = (U, \sigma_U, 1)$ that simulates $T$ using only 2 distinct glue types, tile geometries of size 4$n$, where $n$ is the number of glue types in $T$, and scale factor 1.

Proof To prove Theorem 1, we will begin with an arbitrary temperature-1 aTAM system $T = (T, \sigma_T, G_T, 1)$ which has a symmetric glue function, and we will construct a GTAM system $U = (U, \sigma_U, 1)$ with the necessary constraints and which simulates $T$ at scale factor 1.

The first step in this construction will consist of defining a number of geometries whose compatibility behaviours will emulate the binding behaviours of the glues in $T$. For purposes which will become clear later in the proof, we will define two geometries in $U$ for each glue in $T$. Let \{g$_1$, ..., g$_n$\} be an enumeration of the glues used in $T$ with $n$ being the number of glues. For each glue $g_i$, in this enumeration, we will construct two geometries, $\gamma_{i\alpha}$ and $\gamma_{i\beta}$, both of size 4$n$. These geometries will be divided into four contiguous domains of size $n$. Figure 6 provides an illustration of how the domains are laid out. The two outermost domains, labelled $\alpha_1$ and $\alpha_2$ from left to right, form what will be called the $\alpha$-domains; likewise, the two inner domains, $\beta_1$ and $\beta_2$, make up the $\beta$-domains. It’s important to notice that, when lying on the abutting faces of two adjacent tiles (i.e. North and South, or East and West), the geometries would be reversed from each other, meaning that the $\alpha_1$ domain of one geometry would be overlapping the $\alpha_2$ domain of the other, and the $\beta_1$ domain of one would be overlapping the $\beta_2$ domain of the other. For this reason, when indexing the bumps from 1 to $n$ in each domain, we will number the bump locations in domains $\alpha_1$ and $\beta_1$ from left to right and in domains $\alpha_2$ and $\beta_2$ from right to left. Thus, in Fig. 6, for example, geometry $\gamma_{1\alpha}$ has a bump in location 1 in domain $\alpha_1$ and two bumps in domain $\alpha_2$ in locations 1 and 3.

Each domain in a geometry serves a purpose. Domains $\alpha_1$ and $\beta_1$ are domains whose bumps indicate the index of the glue being represented. For example, when representing glue $g_1$, the geometry $\gamma_{1\alpha}$ will have a single bump at location 1 in domain $\alpha_1$ and the geometry $\gamma_{1\beta}$ will have a single bump at location 1 in domain $\beta_1$. The domains $\alpha_2$ and $\beta_2$ are used to functionally indicate the binding behaviour of the glue being represented. A bump in either of these domains encodes the index of a glue which cannot bind with the glue being represented. For example, in Fig. 6, the glue function $G$, represented as a matrix, indicates that glue 1 cannot bind with either glue 1 or 3, thus the geometries corresponding to $g_1$, namely $\gamma_{1\alpha}$ and $\gamma_{1\beta}$, have bumps in domains $\alpha_2$ and $\beta_2$, respectively, at locations 1 and 3. Notice that, because this glue has no binding strength with itself, if either of these geometries appeared on the abutting faces of two tiles, those
tiles would be incompatible, since a bump at location 1 in domain $\alpha_1$ would intersect with a bump at location 1 in domain $\alpha_2$ and likewise for the $\beta$-domains.

Explicitly, for each glue $g_i$ in $T$, our construction of the geometries $\gamma_{\alpha}$ and $\gamma_{\beta}$ is as follows. Let both the domain $\alpha_1$ in $\gamma_{\alpha}$ and the domain $\beta_1$ in $\gamma_{\beta}$ contain a single bump at location $i$. Then for each $j$ such that $G_T(i, j) = 0$, i.e. $g_i$ and $g_j$ do not bind with any strength, let domain $\alpha_2$ in $\gamma_{\alpha}$ and domain $\beta_2$ in $\gamma_{\beta}$ have a bump at location $j$, remembering to count from right to left in these domains. Everywhere else will contain no bumps in either geometry. Here, it’s important to note that the $\alpha$-domains of $\gamma_{\alpha}$ are identical to the $\beta$-domains of $\gamma_{\beta}$. It may seem as though these geometries are redundant; however, as will be explained later, they become important when trying to simulate certain behaviours of some tile sets. Furthermore, it’s not difficult to see that exactly when two glues in $T$ can bind, as per the glue function $G_T$, the corresponding geometries of a single type, either $\alpha$ or $\beta$, will be compatible with each other. Also notice that $\alpha$ and $\beta$ geometries will always be compatible with each other no matter the corresponding glues since they use non-overlapping domains.

Now let $g^\alpha$ and $g^\beta$ be the 2 glues to be used in $U$, with the property that each binds to itself with strength 1 and not the other. For each tile $t_i \in T$, we will construct 16 different tiles $u_{t_i}^1, \ldots, u_{t_i}^{16}$ in $U$. Each of these 16 tiles will correspond to the single original tile for purposes of simulation, but, as will be seen later, multiple corresponding geometric tile types are necessary in order to allow for adjacent tiles to have mismatching glues, as can happen in situations like that depicted in Fig. 7. If $g_N$ is the glue that appears on the north face of $t_i$, then there are two corresponding geometries that can go on the north face of a corresponding geometric tile in $U$, namely $\gamma_{\gamma_{\alpha}N}$ and $\gamma_{\gamma_{\beta}N}$. Since this is true for all four glues of $t_i$, there are $2^4 = 16$ possibilities for constructing geometric tiles that correspond to $t_i$. Moreover since each side of a geometric tile type needs a glue as well as a geometry, we assign the corresponding glue, $g^\alpha$ or $g^\beta$, to a side depending on whether it has the $\alpha$ or $\beta$ version of the geometry. These 16 $|T|$ geometric tile types make up the tile set $U$.

Because temperature-1 does not allow for cooperative binding between tiles, there are only a few cases that need to be considered to see that $U$ has equivalent dynamics to $T$. Consider the situation in which there is a tile $t_0$ which has a glue, say $g_0$, on its north face, north chosen without loss of generality, that binds with strength 1 during the attachment of one or more tiles, say $t_1, \ldots, t_m$, in $T$. Because $g_0$ binds to these tiles, they must have glues on their south faces which can bind with strength 1 to $g_0$. Given that $T$ is arbitrary, let these be called $g_1, \ldots, g_k$. Now consider the situation in $U$. We can assume that a tile $u_0$, corresponding to $t_0$, has already been placed and that the geometry on its north face is either $\gamma_{\gamma_{\alpha}0}$ or $\gamma_{\gamma_{\beta}0}$. Without loss of generality, we can assume $\gamma_{\gamma_{\alpha}0}$, thus we know, by definition, that the glue on the north face of $u_0$ is $g^\alpha$. Because $u_0$ is in a temperature-1 GTAM system, growth happens exactly when there is a glue match and a compatible geometry. We know that the only geometries that are compatible with $\gamma_{\gamma_{\alpha}0}$ are the $\alpha$ versions of those which correspond to $g_1, \ldots, g_k$, as this is how the geometries were defined, and any $\beta$ version geometry; however, because $u_0$ has glue $g^\alpha$ on its north face, only the $\alpha$ version geometries will be able to attach since any tile face with a $\beta$ version geometry will have glue $g^\beta$.

Therefore, the tile that attaches to the north of $u_0$ will be a tile with an $\alpha$ geometry corresponding to one of the glues $g_1, \ldots, g_k$ on its south face. By definition, these are the tiles corresponding to $t_1, \ldots, t_m$ or, more precisely, the half of the corresponding geometric tiles with an $\alpha$ type geometry on their south sides. If there is no other tile adjacent to this tile being placed, then nothing more needs to be considered, and a tile corresponding to a valid tile in $T$ can be placed north of $u_0$. However, when there is an adjacent tile to the tile being placed, as in the case of Fig. 7, then additional consideration is necessary. This is the reason why 16 tile types corresponding to each tile type in $T$ were necessary. Consider the situation in Fig. 7 where the tile with a single green glue represents $t_0$ and the tile with the single red glue represents a tile adjacent to the tile about to be placed. Even supposing that red glues mismatch with blue glues, the tile with a green and blue glue should be a valid tile to place to the north of $t_0$ since it matches at least one glue in $T$. In $U$ however, if we only had geometries of one type, the geometries corresponding to the red and blue glues would be incompatible and not allow the tile to be placed. Having a second glue type however, allows us to have a version of the tile which can be placed north of $u_0$ and mismatch on the left. Since the $\alpha$ and $\beta$ type geometries are always compatible
with each other, such mismatched glues can always be placed adjacent to each other even though their glues don’t match.

To complete the construction, given \( \sigma_T \), it is straightforward to construct \( \sigma_U \). For each tile in \( \sigma_T \) simply replace it with a corresponding tile in \( \sigma_U \). Since there are 16 such tiles for each tile in \( \sigma_T \), start with the tile using all \( \alpha \) type geometries. Then, if there are any mismatches between adjacent geometries, choose, arbitrarily, one of the mismatched geometries to be a \( \beta \) type. Since this does not affect the matching of any other geometries and since it fixes the problem for the two mismatched geometries, this procedure can be used to fix all mismatched geometries until there are none in \( \sigma_U \).

Finally, to show that \( \mathcal{U} \) simulates \( T \), we first note that for each tile type \( t_i \in T \), the representation function \( R \) simply maps each of the 16 tile types in \( U \) which were explicitly created to represent it (with each permutation of \( \alpha \) and \( \beta \) sides) to \( t_i \). It is then clear that \( R(\sigma_U) = \sigma_T \), and for every tile addition which is possible to \( \sigma_T \), and then to any producible assembly \( \alpha \in A[T] \), tiles of exactly one of the 16 tile types in \( U \) which map to that tile can be added to the corresponding \( \beta \in \bar{A}[U] \) where \( R(\beta) = \alpha \), and no other attachments are possible to \( \beta \). Therefore, \( \mathcal{U} \) and \( T \) have equivalent productions and dynamics, and \( \mathcal{U} \) simulates \( T \). This simulation is done at scale factor 1, and \( U \) has only 2 distinct glues and geometries of size \( 4n \). \( \square \)

This construction demonstrates that the behaviour of temperature-1 aTAM systems with arbitrary symmetric glue functions can be simulated by temperature-1 GTAM systems with no cost in scale using a fixed number of glues, namely 2. It’s important to note that the GTAM systems in our construction only use standard glues which bind only to themselves. This proof implies that two glues in a GTAM system are sufficient for simulating arbitrary aTAM systems at temperature-1; the following theorem shows that, using fewer than two glues, not all aTAM systems can be simulated by GTAM systems. This implies that two glues are necessary for allowing glue mismatches to be properly simulated.

**Theorem 2** There exists a temperature-1 aTAM system which cannot be simulated by a temperature-1 GTAM system at scale factor 1 using \(< 2 \) glues, regardless of the geometry size of the tiles.

**Proof** Consider the temperature 1 aTAM system, say \( T \), presented in Fig. 8, wherein the tile labeled \( S \) is the seed. Notice that \( T \) is not directed as there are multiple terminal assemblies, each of which is a \( 2 \times 2 \) square. Also, let it be the case that each glue binds only to itself, so that a blue glue binds only to a blue glue, for example. Now, for contradiction, suppose that there is some temperature-1 GTAM system \( \mathcal{U} \) that does simulate \( T \) at scale factor 1 using only a single glue. Since \( \mathcal{U} \) simulates \( T \), it must be able to simulate the growth of \( T \) from any possible configuration. Consider then the configuration of \( T \) in which the tiles labelled \( U \) and \( R \) have attached to form an \( L \) shape. In this configuration, there are two tiles which can attach into the corner opposite to tile \( S \): tiles \( A \) and \( B \). Furthermore, notice that if either tile attaches, there will be some glue mismatch since the blue glue does not match with either the red or green glues.

Now imagine a corresponding, \( L \) shaped configuration in \( \mathcal{U} \). Since \( \mathcal{U} \) simulates \( T \), there must be some geometric tile corresponding to either tile \( A \) or \( B \) which can attach in the corner opposite \( S \). If we suppose, without loss of generality, that this was a tile corresponding to \( A \), then it must be the case that this geometric tile has a geometry on its west face which is compatible with the geometry on the east face of a tile corresponding to \( U \) despite the fact that the glues don’t match in \( T \). Therefore, in the case where a geometric tile corresponding to tile \( R \) hasn’t yet attached, there would be the possibility for the geometric tile corresponding to \( U \) to attach to that geometric tile corresponding to \( A \) since there is only a single strength-1 glue type and the geometries are compatible. This, however, would be a violation of the dynamics of \( T \) and thus such a \( \mathcal{U} \) could not simulate \( T \). \( \square \)

The previous proof demonstrated that a GTAM system needs at least two glues to simulate aTAM systems at temperature-1 and scale factor 1. Furthermore, the prior proof gave a construction of a GTAM system which used exactly two glues to simulate any aTAM system at temperature-1 and scale factor 1. Additionally, the construction used geometries of size \( 4n \), and it is shown in Fu et al. (2012) that the lower bound on the size of geometries needed to represent some non-diagonal glue functions is \( \Theta(n) \).

### 4 Simulating higher temperature aTAM systems

In this section, we make use of a generalized version of the GTAM in which it is allowed for each bump location in a
geometry to have its own strength-1 glue rather than having a single glue per geometry. In this model, we can use a very similar strategy to the one above to show that any aTAM system, even one with a temperature parameter greater than 1, with an arbitrary symmetric glue function can be simulated by such geometric tiles using only diagonal glue functions. We call this modified model of geometric tile assembly the sticky Geometric Tile Assembly Model (sGTAM).

**Theorem 3** For any aTAM system \( T = (T, \sigma_T, G_T, \tau) \) with arbitrary symmetric glue function \( G_T \), there exists a temperature-\( \tau \) sGTAM system \( \mathcal{U} = (U, \sigma_\mathcal{U}, \tau) \) that simulates \( T \) using only 1 glue type, geometries of size \( n \tau + 2 \) where \( n \) is the number of glue types in \( T \), and at scale factor 1.

**Proof.** Let \( T = (T, \sigma_T, G_T, \tau) \) be an arbitrary aTAM system where \( G_T \) is a symmetric glue function. We will construct an sGTAM system \( \mathcal{U} = (U, \sigma_\mathcal{U}, \tau) \) so that \( \mathcal{U} \) simulates \( T \) at scale factor 1. Like the temperature-1 construction above, this construction makes use of geometries which are divided into domains so that each location in a domain corresponds to a unique glue type in \( T \). Unlike the construction above, however, the number of domains per geometric tile is equal to the temperature \( \tau \) rather than being fixed at 4. Moreover, we will not need to use inner and outer versions of identical geometries since, in our modified model, each location in a domain can have its own glue, and by choosing these locations carefully, we can allow for pairs of sticky geometries which do not bind with any strength.

The construction of \( \mathcal{U} \) is as follows. First, we define glue type \( g_U \) (which binds to itself with strength 1) to be used on specific locations of the sticky geometries of tiles in \( U \), and all other locations have the null glue. Then, let \( \{ g_1, \ldots, g_n \} \) be an enumeration of the geometries used in \( T \). For each glue \( g_i \) in this set, we construct two corresponding sticky geometries (exactly one of which will be on each side of each tile in \( U \)). One of these geometries, which we call \( \lambda_i^{\text{index}} \), encodes the index of the \( g_i \), namely \( i \). The other geometry, which we call \( \lambda_i^{\text{bind}} \), encodes the index of the \( g_i \) binds to and the corresponding strength. Each of these geometries is of size \( n \tau + 2 \) and is divided into \( \tau \) domains of size \( n \) with two additional bump locations on the far left and far right. Each of the \( \tau \) domains corresponds to a binding strength less than or equal to \( \tau \). In each domain of \( \lambda_i^{\text{bind}} \), the \( i \)th bump location will have glue \( g_U \); all other locations will have no glue (i.e., the null glue). For integer \( k \leq \tau \), the \( j \)th location in the \( k \)th domain of \( \lambda_i^{\text{bind}} \) will have glue \( g_j \) if and only if \( g_i \) binds to \( g_j \) with strength greater than or equal to \( k \); otherwise, it will have no glue. Additionally, the index-geometries will have bumps only in the far left and far right bump locations, whereas the bind-geometries will have bumps in every location except for the far left and far right. An example of what these sticky geometries would look like, given an example glue function for some aTAM system, is illustrated in Fig. 9.

For each tile type \( t \in T \), we construct a single corresponding tile type \( u \in U \) such that if \( t \) is defined by the glue functions \( \{ g_N, g_E, g_S, g_W \} \) where \( N, E, S, W \) are the cardinal directions, then \( u \) is defined by the sticky geometries \( \{ \lambda_N, \lambda_E, \lambda_S, \lambda_W \} \). Notice that the North and East geometries are index types whereas the South and West are bind types. This ensures that when two tiles are adjacent, their abutting sticky geometries will consist of a single index type and a single bind type. To see why this simulation preserves the dynamics of \( T \), consider two tiles \( t_1, t_2 \in T \) and their corresponding tiles \( u_1, u_2 \in U \). Suppose that \( t_1 \) is attaching above \( t_2 \) and that \( t_1 \)’s South glue, say \( g_i \), binds to \( t_2 \)’s North glue, say \( g_j \), with some strength \( k \leq \tau \). In this case, \( u_1 \)’s South geometry will be \( \lambda_i^{\text{bind}} \) and \( u_2 \)’s North geometry will be \( \lambda_j^{\text{index}} \). Since \( \lambda_j^{\text{index}} \) only has strength 1 in the \( j \)th location of each of its \( \tau \) domains and \( \lambda_i^{\text{bind}} \) has strength 1 in the \( j \)th locations of each of only \( k \) of its \( \tau \) domains, \( u_1 \) and \( u_2 \) attach with strength \( k \). This argument easily generalizes to all directions and it is not difficult to see that cooperative dynamics are preserved since we are using scale factor 1.

It should be noted that the physical bumps are only used to ensure that geometries interlock and to prevent tiles from attaching skewed to one another which might allow for tiles to attach with more or less strength than they should.

5 Duple tiles cannot be simulated at temperature-1 using scale factor 1

As mentioned previously, another model of tile assembly which makes use of geometric hindrance is the Duplicated abstract Tile Assembly Model (DaTAM). This model makes use of \( 2 \times 1 \) (or \( 1 \times 2 \)) domino tiles called duples in addition to the normal square tiles. Similar to the GTAM, this model is
capable of universal computation at temperature 1 by means of geometric hindrance. Given that both geometric tiles and duple tiles have the capacity to block tiles from attaching in an adjacent location and that geometric tiles can simulate normal tiles with scale factor 1 at temperature 1, a natural assumption would be that geometric tiles could simulate duple tiles under these same constraints, using two geometric tiles to represent a single duple tile. This however, is not the case. The situation where it’s possible for two duple tiles to grow independently in such a way that they would overlap cannot be simulated by geometric tiles using scale factor 1 at temperature 1.

Normally, simulation is defined in terms of a representation function which maps macrotile regions in the simulating system to individual tiles in the simulated system. Since we are using a scale factor of 1, the input to this function would normally be a single tile; however, in order to accommodate the fact that we are dealing with the simulation of a DaTAM system, we relax the restriction and instead allow $R$ to also consider the 4 neighboring tile locations as well (illustrated in Fig. 10) since duple tiles may be simulated using a pair of adjacent geometric tiles. We also allow fuzz of size 2 (illustrated in Fig. 11). These accommodations are rather strong (because, for example, they allow for using tiles in fuzz regions, which don’t map to tiles in the simulated system, to influence how $R$ maps tiles within the assembly), but are intended to make the result stronger and more general. Additionally, as is common with definitions of simulation in tile assembly, we require that empty space cannot map to a square (i.e. non-duple) tile and that at least one of the two locations corresponding to a duple must contain a tile in order for those locations to resolve to a duple under $R$. Obviously, it is also required that if a location corresponding to one half of a duple maps to a duple tile, the location corresponding to the other half must also map to that duple (although we allow for at most one of those locations to be empty). Even with these accommodations, we show that it is impossible for all systems in the DaTAM to be simulated by systems in the GTAM.

Before we state and prove the main result of this section, we first state and prove a useful lemma.

**Lemma 1 (The Surrounded Fuzz Lemma)** Let $D$ be a DaTAM system and $\mathcal{G}$ be a GTAM system which simulates $D$ under representation function $R$ and at scale factor 1. Given an assembly $\alpha \in \mathcal{A}[D]$, let $S \subseteq \mathbb{Z}^2$ be a region of the plane such that (1) $S$ is completely surrounded by tiles of $\alpha$, (2) all locations in $S$ are empty, (3) no location in $S$ is adjacent to an empty location which is not in $S$, and (4) there exists a valid assembly sequence in $D$ starting from $\alpha$ such that every point of $S$ receives a singleton tile. For each $\beta \in \mathcal{A}[\mathcal{G}]$ such that $R(\beta) = \alpha$, the number of empty locations in $S$ surrounded by $\beta$ must be greater than or equal to $|S|$.

**Proof.** Suppose the number of empty locations in $S$ surrounded by $\beta$ was less than $|S|$ and let $\bar{\alpha}$ be an assembly sequence in $D$ such that, starting from $\alpha$, every point in $S$ receives a singleton tile (in some fixed order). By the definition of simulation of DaTAM systems by GTAM systems, it is required that $R$ cannot map empty space to a singleton tile and thus that for every tile attachment at a location $l$ in $\bar{\alpha}$, the corresponding assembly sequence, say $\bar{\beta}$, in $\mathcal{G}$ which models $\bar{\alpha}$ (which must exist) must also place a tile at location $l$. (Keep in mind that during $\bar{\beta}$ other tile attachments may occur as well). Additionally, the definition of simulation, specifically that $\mathcal{G}$ models $D$, forbids the placement of a single tile in $\mathcal{G}$ from changing the representation of its assembly, under $R$, by the addition of more than one tile, otherwise there would never be an assembly which maps to the intermediate assembly in which only one of the tiles has attached. Thus, for each singleton tile attachment at location $l$ in $\bar{\alpha}$ there must exist a singleton tile attachment at $l$ in $\bar{\beta}$; however, by the pigeonhole principle, since the number of empty locations at which tiles can attach in $S$ surrounded by $\beta$ is less than the number...
We prove Theorem 4 by contradiction. Therefore, assume that GTAM system $G = (G, \sigma_D, \tau)$ simulates $D$. We will now describe a series of assembly sequences in $D$, each of which must be modeled by $G$, which will cause us to arrive at a contradiction. First, consider the assembly sequence $\alpha$ in $D$ wherein all of the tiles labeled $B$ through $P$ attach in clockwise order to the seed $A$ tile to form a square as illustrated in Fig. 13. (Note that these attachments could happen in many orders, and we’ve arbitrarily chosen this ordering for $\alpha$.) Then consider any corresponding assembly sequence $\alpha'$ in $G$, i.e. $R$ maps the assemblies produced by $\alpha'$ to those produced by $\alpha$, in order. Note that such an assembly sequence must exist if $G$ simulates $D$.

Because of the allowance of fuzz in $G$, and the larger domain of the representation function $R$, during $\alpha'$ it might be the case that tiles attach in locations which, even after their attachment, still map to empty space (i.e. are fuzz, meaning that for such a tile, its location $l$ is not in the domain of $R$ and thus it does not represent a tile of $D$) after the square has assembled. Note though that this cannot happen inside of the square by Lemma 1 and the fact that a valid assembly sequence which extends $\alpha$ in $D$ could be that which produces the assembly in Fig. 14, where the center of the square is completely filled by singleton tiles, i.e. the $S$ of Lemma 1 equals the points 1 through 9 of Fig. 13. We then instead consider the extension of assembly sequence $\alpha$ in $D$ where the duple tile $X$ attaches in locations 4 and 5 (as labeled in Fig. 13). Consider the cases where, in the simulation of this attachment in $G$, the tile in location 5, corresponding to the right half of the duple, is either the first or the only of the two halves to attach in $G$, at the first point when $R$ maps it to a duple of type $X$. In these cases, notice that there must be a fuzz tile, say $t$, in at least one of the locations 2, 6, or 8. (Otherwise, the tile in location 5 would not be attached to the assembly.) Without loss of generality suppose that $t$ attached in location 8. In the case where $R$ maps the tile in $5$ to a duple $X$ before the attachment of a tile in location 4, consider the extension of $\alpha$ (following the attachment of the duple $X$) where $W$ tiles attach in locations 7 and 9. The corresponding assembly sequence in $G$ must have tiles placed in those locations by the definition of simulation. Notice though that in this case, $t$ is surrounded by tiles to its north, south, east, and west. Because of this, if $R$ maps $t$ to empty space, $t$ will remain mapped to empty space forever, which is invalid since in $D$ a tile must still attach there before the assembly can be terminal; however, if $R$ maps $t$ to a tile, then the simulation does not correctly simulate the assembly sequence where just the tiles in locations 7 and 9 attach, before a tile attaches in location 8. This contradiction shows that the tile in location 5 cannot be the only tile to attach before being mapped to the duple $X$.
Now consider the case in $G$ where a tile in location 5 attaches first, but a tile in location 4 also attaches before $R$ maps them to the duple $X$. In this case, we still must have at least one fuzz tile $t$ which attached prior to the tile in location 5, and without loss of generality again assume that is in location 8. By letting $S$ of Lemma 1 be the set of locations \{1, 2, 3, 6, 7, 8, 9\} and again extending $\tilde{a}$ so that it fills those locations with singleton tiles of type $X$ in the numerical order of the locations (this is a valid assembly sequence in $D$ starting from our current assembly which has all of the tiles in Fig. 13 plus the duple $X$ in locations 4 and 5) we see that this is also impossible by Lemma 1, since there would be fewer open locations of $S$ in $G$.

So, now let the assembly sequence $\tilde{a}_X$ in $D$ be that wherein all of the tiles labeled $B$ through $P$ attach in clockwise order to the seed $A$ tile to form a square as illustrated in Fig. 13, followed by the attachment of a duple of type $X$ in locations 4 and 5. Let $\tilde{\beta}_X$ be a minimal (fewest tile attachments after the growth of the square) corresponding assembly sequence in $G$ (which again must exist to be a valid simulation). As we have shown, it must be the case that a tile attaches in location 4 before any tile attaches in location 5. First notice that before the attachment of a tile in location 4 during $\tilde{\beta}_X$, it cannot be the case that any tile attaches in locations 1 or 7. Supposing such a tile attachment did occur, say at location 1 without loss of generality, we could consider the assembly sequence in which a $Y$ tile attaches at location 2 in $D$. Since the tile at location 1 in $G$ would be surrounded on all sides it must map, under $R$ to either empty space or a tile. By the previous argument, both cases represent invalid simulations. Additionally, it cannot be the case that, during $\tilde{\beta}_X$ before a tile is placed at location 4, a tile attaches in locations \{2, 3, 6, 8, 9\}. Such attachments, as shown previously, would have to occur after the square formed and cannot affect the attachment of the tile in location 4 and since we required that $\tilde{\beta}_X$ be the minimal corresponding assembly sequence, these attachments cannot occur. (Recall that we’ve already ruled out the cases where a tile could attach in location 5 before one in 4.) Thus for the assembly sequence $\tilde{a}_X$ in $D$ there exists an assembly sequence in $G$ during which, immediately after the growth of the square, a tile is placed at location 4. A symmetrical argument shows if we considered the attachment of the duple $Z$ in $D$ in locations 5 and 6, corresponding assembly sequences in $D$ and $G$, say $\tilde{a}_Z$ and $\tilde{\beta}_Z$ respectively, exist for a tile in location 6.

Now consider the assembly sequence $\tilde{\beta}_{XZ}$ which consists of all tile attachments of $\tilde{\beta}_X$ until the attachment of the tile in location 4 and then the attachment of the tile in location 6 from $\tilde{\beta}_Z$, which must be a valid assembly sequence since it only requires the binding of two completely independent tiles which were previously shown to be able to attach in those locations. There are a few possible cases to consider. If each of the tiles in locations 4 and 6 map, under $R$, to their corresponding duple tiles (of types $X$ and $Z$), the simulation is invalid since only one duple can attach in the center row. Thus, it must be the case that only one or neither of the duple tiles are represented. Without loss of generality suppose that the $X$ duple tile has not yet resolved and will not resolve, and consider the extension to the assembly sequence where a $Y$ tile attaches in location 2 and a $W$ tile attaches in location 8 in $D$. If we consider the corresponding assembly sequence in $G$, notice that it must be the case that, if location 5 is currently empty but should a tile be able to attach there, it would be able to now. If that center tile can attach, suppose that it does in $G$. Then consider the assembly sequence where a $Y$ tile attaches in location 1 and a $W$ tile attaches in location 7 in $D$. Notice then that in the corresponding assembly sequence in $G$, the west half of the duple $X$ will be surrounded (or have an empty location to its right which cannot be filled) and, by definition of the representation function $R$ must either resolve to a tile, which is invalid since the corresponding tile has not yet attached in $D$, or to empty space, which is invalid since it must then permanently remain empty space whereas a tile can eventually attach in $D$. Therefore, in all cases, $G$ fails to simulate $D$. Thus, $D$ is not simulated by $G$, and therefore Theorem 4 is proven.

6 Glue cooperation cannot be simulated with geometric Hindrance

In this section, we prove that there exists a directed temperature-2 aTAM SASS (i.e. a fully deterministic temperature-2 aTAM system) which cannot be simulated by any temperature-1 GTAM system. We first provide a brief high-level overview of our proof, and then the full proof.

Theorem 5 There exists a directed temperature-2 aTAM SASS $S$ that cannot be simulated by any GTAM system at temperature 1.

Figure 15 shows a schematic drawing of a temperature-2 SASS $S$ which cannot be simulated. Essentially, it grows a “planter” module (similar to that of Lathrop et al. (2011)) to form an infinite assembly growing to the right, that initiates an infinite series of counters that grow upward to every height $\geq 4$. (Figure 16a shows the pattern of growth which allows $S$ to be a SASS.) Each counter then grows an arm down which crashes into a portion of the assembly below it, but since the arms grow longer and longer, eventually they reach a point where they must “pump”, or grow in a periodic manner. However, in order to correctly grow macrotiles which simulate the cooperative growth between the end of each arm and the bottom portion of the assembly, in the simulating system there must be a path of tiles which can grow out from each arm. Since the arms must become periodic, those paths could also
To prove Theorem 5, we first present the details of the
Examples can be seen in Figs. 16b and 17.

Fig. 15 Overview of a temperature-2 aTAM system which cannot be
simulated by any temperature-1 GTAM system at scale factor 1.

Fig. 16 (a) Depiction of one iteration of the growth of T, with arrows
showing the ordering of growth. (b) Zoomed in portion of the con-
struction shown in Fig. 15 which shows (with a solid line) an example of
a path of tiles bound by glues which must extend from a tile, a, in the
supertile representing a green tile, to a tile, b, in the supertile represen-
ting the red tile. The dashed line shows how a previous copy of a could
allow growth of the same path in a higher location.

Fig. 17 (Left and middle) Examples of windows, w and w’ which each
cut a portion of the supertiles representing the green column, plus the
yellow and red tiles, from the rest of an iteration, and which have
identical glue bindings across them. Note that glue bindings only occur
across the top line of each, and the bottom line separating the inside
from the planter goes only between unbound tiles, (right) Assembly
α_Lβ_R (where β_R is simply a translated copy of β_R) which must be able
to form by the Window Movie LemmaMeunier et al. (2014). Even if
the representation of the red tile isn’t complete, the allowed boundary
for the growth of fuzz is broken.

grow in higher locations, which leads to invalid simulation.
Examples can be seen in Figs. 16b and 17.

We now present the details of the proof of Theorem 5.

Proof To prove Theorem 5, we first present the details of the
aTAM SASS S, and then explain why no GTAM system
at temperature-1 can simulate it. The temperature-2 aTAM
SASS system S grows as follows. (Reference Figure 15 for
a graphical depiction.) From the seed tile (shown as a black
square on the far left), a module called the planter (simi-
al to the planter module introduced in Lathrop et al.
(2011)) grows horizontally to the right (white portion). At a
high level, the planter is a binary counter which counts
from 4 to infinity, with spacing between each increment that
allows the bits of the current binary number to be rotated
to the north, as well as 6 additional spaces. At each location
where the bits of a binary number b are exposed to the north, a
binary decremrenter grows upward (light grey rectangles
in Fig. 15) to a height of 2b by having each successive pair
of rows decrement the values from b − 1 to 0, then check to
see if 0 has been reached. Once the decremrenter reaches
0 and terminates upward growth, a row of tiles grows east
from its top row, a distance of 4 tiles. Then, from the east-
ernmost tile of that row, a column of tiles (green in Fig.
15) grows downward until being blocked by the planter
(shown in more detail in Fig. 16a). At that point, coopera-
tion between the final green tile and a tile in the planter
allow for placement of the yellow tile, which then exposes
a strength-2 glue allowing the red tile to attach. We call the
growth of a decremrenter through the placement of its
associated red tile an iteration, and note that the in-
finite terminal assembly of S grows exactly one iteration
for every number ≥ 4. (Note that the counter within the
planter, the decremrenter module, etc. all use stan-
dard, basic aTAM modules.)

As depicted in Fig. 16a, the growth of the modules of
an iteration in general follows a zig-zag pattern which
allows the growth to be completely sequential, only ever
having a single frontier location, and thus maintaining
the property that S is a SASS. Additionally, the growth of the
planter extends 5 columns beyond the east side of its asso-
ciated decremrenter before backtracking and growing the
top row, which ensures that the tiles that block and cooperate
to place the green and yellow tiles, respectively, are in place
before the green column grows downward. Thus, clearly, S
is directed, and since at every point during its growth there
is exactly one single frontier location, S is a SASS.

We prove Theorem 5 by contradiction, and therefore
assume that some temperature-1 GTAM system G correctly
simulates S under some representation function R, and let m
be the scale factor of that simulation. Without loss of gener-
ality, we will assume that the tiles of G contain no glues to which
no other tile can bind, as any such glues could be removed
without changing the behavior of G. Now, we note that the
height of each successive decremrenter, and therefore the
heights of the associated green columns, increases infinitely.
However, G must have a tile set with a finite number of glue
types. Let g be the number of glue types in G and for our
proof we’ll let n = ((g + 1)^(6m) · (6m)! + 1) · 3 + 2 (we’ll
discuss the selection of this value later) and we’ll focus on
the growth of an assembly produced by G up to and including
the first tile placement of iteration n in which the red
tile of iteration n of S is represented under R (depicted
in Fig. 16b as b). Let α be the assembly created at this point,
and note that by the assumption that G simulates S, such an α
must be producible in \( \mathcal{G} \). Furthermore, we will define \( \alpha = (\alpha_i \mid 0 \leq i < k) \) to be the assembly sequence which produces \( \alpha \) and give it the following restriction: whenever there are multiple frontier locations in any \( \alpha_i \) (since \( \mathcal{G} \) itself need not be a SASS even though \( S \) is), the next location to receive a tile will be selected from those with the lowest \( y \)-coordinate. This would clearly be a valid assembly sequence, and since \( \mathcal{G} \) is simulating \( S \), which is a SASS and grows each portion of each iteration in the previously specified order, it must be the case that no locations further than the single supertile boundary of fuzz outside of the first \( n \) iterations can receive a tile before the \( n \)th iteration completes. Additionally, selection of the assembly sequence ensures that all tiles which share a path in the binding graph (i.e. a path through a series of bound glues) that includes a tile in the planter without going through the full decrementer and green column (i.e. those which may have grown upward as fuzz from the planter), will be placed before the second row of the decrementer completes growth.

The tile placement which causes the red tile of iteration \( n \) to be represented under \( R \) (shown as \( b \) in Fig. 16b) must be placed only after representations of the entire decrementer, the green column, and the yellow tiles have grown. This means that it must have a path in the binding graph which connects upward through the supertiles representing the green column, since the assembly sequence was chosen to place any tiles which didn’t have such a path before the decrementer completes, and if a tile placement which caused the red tile to be represented were placed before growth of the green column, then \( \mathcal{G} \) would not correctly simulate \( S \) (which can only place the red tile after the decrementer and green column complete). We now know that there must be a path of bound glues connecting the tile at location \( b \) to some tile within the green column (which we label as \( a \) in Fig. 16b).

We will now consider cuts which separate the green column in \( \alpha \) into two pieces and cross between macrotiles (of width \( m \)) representing a green tile, as well as possibly one macrotile of allowable fuzz on each side, for a maximum width of \( 3m \). The count of the maximum possible number of glue positions along that cut from tiles both above and below it is then \( 6m \). With \( g \) glue types and the empty glue, there are \((g + 1)^{6m}\) different ways to fill those positions. The total possible number of orderings of placements at each of those \( 6m \) positions is \((6m)!\), and therefore \((g + 1)^{6m} \cdot (6m)! + 1\) is greater than the total possible variety of glues and orderings of their placements along the cut. The quantity \( n \) represents the number of the final iteration of \( S \) which is being represented in \( \mathcal{G} \), and thus the height of the column of green tiles in \( S \), and macrotiles representing those tiles in \( \mathcal{G} \), will be \( n + 4 \). By setting \( n = ((g + 1)^{6m} \cdot (6m)! + 1) \cdot 3 + 2 \), we can ignore the top and bottom three green macrotiles and still guarantee that even if we only inspect the cuts between every third pair of macrotiles, that at least two of those cuts have the same window movie (i.e. they had the exact same glues placed in the exact same order). Now, using those two cuts we define the windows \( w \) and \( w' \) (exemplified in Fig. 17) such that they each have one of those identical cuts as their northern horizontal cut, their western edge travels down between any allowable fuzz macrotiles (and there must be at least one completely empty macrotile space between the decrementer and the green column or the fuzz boundary would be broken), and then back to the east in a perhaps jagged path which avoids crossing any bound glues but separates the portion of \( \alpha \) attached to the green column from the planter. This type of cut must be possible because, as previously mentioned, the assembly sequence was selected to first place all tiles which could be bound by glues through a path directly to the planter (avoiding the green column) before the green column forms, and since \( \mathcal{G} \) is temperature-1, if tiles could bind after growing down from the green column, they could also bind before it forms. Thus, the boundary of the assembly on the north side of the planter which forms before the green column can be followed for this cut, to finish the windows.

Note that \( w' \) is not a full translation of the window \( w \) by some vector \( c \), but instead only the upper horizontal cut which contains all locations which could have a glue bond (all other locations along the window cut were chosen so that no glue bonds were formed across them). By Lemma 3.3 (the Window Movie Lemma) of Meunier et al. (2014), and more specifically Corollary 3.4 which refers only to the bond-forming submovies, it must be the case that the assembly depicted on the right of Fig. 17 must be able to form. Essentially, the existence of two cuts across the green column where identical series of glues are placed along those cuts allows the segment in between those cuts to be “pumped”, potentially either upward (increasing the number of occurrences of the subassembly in between) or downward (decreasing them). Here we choose to pump down and remove the intermediate subassembly, showing that the path of tiles which grows to the right from the green column must be able to also grow at a higher location. Since, as we have shown, this path must extend greater than the width of a single supertile beyond the green column (i.e. through the width of the yellow macrotile and into the red), it is beyond the allowable fuzz region when translated upward, and therefore \( \mathcal{G} \) does not correctly simulate \( S \). This is a contradiction that \( \mathcal{G} \) simulates \( S \), and therefore Theorem 5 is proven.

While Theorem 5 states that temperature-1 GTAM systems cannot even simulate the full class of directed temperature-2 aTAM single-assembly-sequence systems, the following

---

5 We’ll refer directly to the arXiv version here [https://arxiv.org/pdf/1304.1679.pdf](https://arxiv.org/pdf/1304.1679.pdf) for convenience.
result generalizes that to show that the same is true across all systems relying on weak cooperation across any tile assembly model.

**Theorem 6** There exists a directed temperature-2 aTAM SASS $S$ that cannot be simulated by any weakly-cooperative tile assembly system that relies on geometric hindrance.

**Proof** The proof of Theorem 6 follows almost immediately by using an identical proof structure to the proof of Theorem 5. Let $S$ be the same as $S$ defined in that proof. Again, our proof will be by contradiction, and we will therefore assume that $\mathcal{W}$ is a weakly-cooperative tile assembly system which utilizes geometric hindrance and simulates $S$. By the definition of simulation, all of the same fundamental restrictions that held for $G$ of the former proof also hold for $\mathcal{W}$, regardless of the model in which it is contained. There must be a bound on the number of glues $g$ (and tile types), and a grid of macrotiles bound to some scale factor $m$, and we can similarly compute a bound on the number of window movies that are possible across the boundaries of the macrotiles representing the green tiles of $S$. In some other models the tiles may be larger than single unit squares (e.g. polyominoes-Fekete et al. (2015)), or they may consist of multiple shapes (e.g. squares and duples-Hendricks et al. (2015)), or they may be able to meet at differing angles (e.g. polygons-Gilbert et al. (2016)), and in such systems the tiles binding across macrotile boundaries may not form straight lines along the macrotile boundaries, but instead may create jagged boundaries. However, even though this may increase the number of window movies possible, they are still finitely bounded. Therefore, we can once again find a pair of windows with identical bond-forming submovies, as well as a path in the binding graph from the green macrotiles to the red. Once again, we can use the Window Movie Lemma to pump down between the top cuts of the windows, and $\mathcal{W}$ must grow that path at a higher location, therefore breaking the boundary allowed for fuzz. This contradicts the assumption that $\mathcal{W}$ simulates $S$, and therefore Theorem 6 is proven. $\square$

Note that Theorem 6 is proven for weakly-cooperative systems using geometric hindrance, but that this does not include the second category of types of binding hindrance, namely systems which use glue repulsion. Although such systems make possible dynamic behavior in which portions of an assembly may break off, and this may make the proof more difficult, we conjecture that they also cannot simulate $S$.

**References**

Cook M, Fu Y, Schweller RT (2011) Temperature 1 self-assembly: Deterministic assembly in 3D and probabilistic assembly in 2D.

In: SODA 2011: Proceedings of the 22nd Annual ACM-SIAM Symposium on Discrete Algorithms. SIAM

Demaine ED, Demaine ML, Fekete SP, Patitz MJ, Schweller RT, Winslow A, Woods D (2014) One tile to rule them all: Simulating any tile assembly system with a single universal tile. In: Proceedings of the 41st International Colloquium on Automata, Languages, and Programming (ICALP 2014), IT University of Copenhagen, Denmark, July 8-11, 2014, LNCS, vol. 8572, pp. 368–379

Doty D, Kari L, Masson B (2013) Negative interactions in irreversible self-assembly. Algorithmica 66(1):153–172

Doty D, Patitz MJ, Summers SM (2011) Limitations of self-assembly at temperature 1. Theor Comput Sci 412:145–158

Fekete SP, Hendricks J, Patitz MJ, Rogers TA, Schweller RT (2015) Universal computation with arbitrary polyomino tiles in non-cooperative self-assembly. In: Proceedings of the Twenty-Sixth Annual ACM-SIAM Symposium on Discrete Algorithms (SODA 2015), San Diego, CA, USA January 4-6, 2015, pp. 148–167. https://doi.org/10.1137/1.9781611973730.12. http://epubs.siam.org/doi/abs/10.1137/1.9781611973730.12

Fu B, Patitz MJ, Schweller RT, Sheline R (2012) Self-assembly with geometric tiles. In: Proceedings of the 39th International Colloquium on Automata, Languages and Programming, ICALP, pp. 714–725

Gilbert O, Hendricks J, Patitz MJ, Rogers TA (2016) Computing in continuous space with self-assembling polygonal tiles. In: Proceedings of the Twenty-Seventh Annual ACM-SIAM Symposium on Discrete Algorithms (SODA 2016), Arlington, VA, USA January 10-12, 2016, pp. 937–956

Hader D, Patitz MJ (2019) Geometric tiles and powers and limitations of geometric hindrance in self-assembly. In: Proceedings of the 18th Annual Conference on Unconventional Computation and Natural Computation (UCNC 2019), Tokyo, Japan June 3–7, 2019, pp. 191–204

Hendricks J, Patitz MJ, Rogers TA (2016) Doubles and negatives are positive (in self-assembly). Nat Comput 15(1):69–85. https://doi.org/10.1007/s11047-015-9513-6

Hendricks J, Patitz MJ, Rogers TA, Summers SM (2015) The power of duples (in self-assembly): It’s not so hip to be square. Theor Comput Sci. https://doi.org/10.1016/j.tcs.2015.12.008

Lathrop JI, Lutz JH, Patitz MJ, Summers SM (2011) Computability and complexity in self-assembly. Theor Comput Syst 48(3):617–647

Lathrop JI, Lutz JH, Summers SM (2009) Strict self-assembly of discrete Sierpinski triangles. Theor Comput Sci 410:384–405

Luchsinger A, Schweller R, Wylie T (2018) Self-assembly of shapes at constant scale using repulsive forces. Nat Comput. https://doi.org/10.1017/10070-1147-1-9707-9

Luchsinger A, Schweller RT, Wylie T (2017) Self-assembly of shapes at constant scale using repulsive forces. In: UCNC, Lecture Notes in Computer Science, vol. 10240, pp. 82–97. Springer

Mennier P, Woods D (2017) The non-cooperative tile assembly model is not intrinsically universal or capable of bounded Turing machine simulation. In: Proceedings of the 49th Annual ACM SIGACT Symposium on Theory of Computing, STOC 2017, Montreal, QC, Canada, June 19-23, 2017, pp. 328–341. https://doi.org/10.1145/3055399.3055446. http://doi.acm.org/10.1145/3055399.3055446

Mennier PE, Patitz MJ, Summers SM, Theyssier G, Winslow A, Woods D (2014) Intrinsic universality in tile self-assembly requires cooperation. In: Proceedings of the ACM-SIAM Symposium on Discrete Algorithms (SODA 2014), (Portland, OR, USA, January 5-7, 2014), pp. 752–771

Patitz MJ, Schweller RT, Summers SM (2011) Exact shapes and Turing universality at temperature 1 with a single negative glue. In: Proceedings of the 17th international conference on DNA computing and molecular programming, DNA’11, pp. 175–189
Reif JH, Sahu S, Yin P (2005) Complexity of graph self-assembly in accretive systems and self-destructible systems. In: International Workshop on DNA-Based Computers, pp. 257–274. Springer
Rothemund PWK, Winfree E (2000) The program-size complexity of self-assembled squares (extended abstract). In: STOC ’00: Proceedings of the thirty-second annual ACM Symposium on Theory of Computing, pp. 459–468. ACM, Portland, Oregon, United States https://doi.org/10.1145/335305.335358

Soloveichik D, Winfree E (2007) Complexity of self-assembled shapes. SIAM J Comput 36(6):1544–1569
Winfree E (1998) Algorithmic self-assembly of DNA. Ph.D. thesis, California Institute of Technology

Publisher’s Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.