Resiliency to Multiple Nucleation in Temperature-1 Self-Assembly

Matthew J. Patitz[1]*, Trent A. Rogers[1]**, Robert T. Schweller[2]***, Scott M. Summers[3], and Andrew Winslow[4]

[1] Department of Computer Science and Computer Engineering, University of Arkansas
patitz@uark.edu,tar003@uark.edu

[2] Department of Computer Science, University of Texas–Rio Grande Valley
robert.schweller@utrgv.edu

[3] Computer Science Department, University of Wisconsin–Oshkosh
summerss@uwosh.edu

[4] Département d'Informatique, Université libre de Bruxelles
awinslow@ulb.ac.be

Abstract. We consider problems in variations of the two-handed abstract Tile Assembly Model (2HAM), a generalization of Erik Winfree's abstract Tile Assembly Model (aTAM). In the latter, tiles attach one-at-a-time to a *seed*-containing assembly. In the former, tiles aggregate into *supertiles* that then further combine to form larger supertiles; hence, constructions must be robust to the choice of seed (nucleation) tiles. We obtain three distinct results in two 2HAM variants whose aTAM siblings are well-studied.

In the first variant, called the *restricted glue 2HAM* (rg2HAM), glue strengths are restricted to -1, 0, or 1. We prove this model is Turing universal, overcoming undesired growth by breaking apart undesired computation assembly via repulsive forces.

In the second 2HAM variant, the *3D 2HAM* (3D2HAM), tiles are (three-dimensional) cubes. We prove that assembling a (roughly two-layer) $n \times n$ square in this model is possible with $O(\log^2 n)$ tile types. The construction uses "cyclic, colliding" binary counters, and assembles the shape non-deterministically. Finally, we prove that there exist 3D2HAM systems that only assemble infinite *aperiodic* shapes.

1 Introduction

Self-assembly is the process through which a group of discrete components combine according to simple and local interaction rules to form a complex final

* This author's research was supported in part by National Science Foundation Grant CCF-1422152.

** This author's research was supported by the National Science Foundation Graduate Research Fellowship Program under Grant No. DGE-1450079, and National Science Foundation Grant CCF-1422152.

*** This author's research was supported in part by National Science Foundation Grants CCF-1117672 and CCF-1555626.

structure. Taking inspiration from the many examples of self-assembly exhibited in nature, researchers are now investigating the use of nanoscale self-assembly for systematic nano-fabrication of atomically-precise computational, biomedical, and mechanical devices.

In the early 1980s, Ned Seeman [21] developed an experimental technique for such fabrication known as *DNA tile self-assembly.* In DNA tile self-assembly, a small number of single strands of DNA are used to form a DNA *tile* with four "sticky ends" consisting of short sequences of unpaired nucleotides, one for each of the the cardinal directions: north, east, south, and west. A sticky end of one tile binds with a sticky end of a second tile if their nucleotides are Watson-Crick complements; more generally, multi-tile *supertiles* bind together similarly. Careful design of sticky ends enables an experimenter to program sets of DNA tiles to self-assemble into target structures.

Erik Winfree's abstract Tile Assembly Model (aTAM) is a discrete mathematical model of DNA tile self-assembly [24]. The aTAM abstracts DNA tiles are as translatable, but not rotatable, square tiles whose sides have alphanumerically labeled *glues* with integer *strengths.* Two tiles or assemblies placed adjacently *bind* if the sums of the strengths of matching glues on coincident sides is at least a specified minimum threshold, called the *temperature* of the system. Self-assembly in the aTAM starts from a unique *seed* tile type and proceeds nondeterministically and asynchronously by single-tile addition to the growing *seed assembly.*

Two-handed tile assembly. A well-studied seedless generalization of the aTAM is the *two-handed abstract Tile Assembly Model (2HAM).* In the 2HAM, growth does not begin at a unique seed tile type. Instead, all possible pairs of tiles bind, followed by all possible pairs of *supertiles*, until no pair of resulting supertiles can bind further.

The role of temperature in both aTAM and 2HAM systems is critical, as it determines the criteria by which supertiles bind: a higher temperature defines a stronger binding criterion. At temperature 2, *cooperative* binding can be used to synchronize assembly and is known to confer complex algorithmic behavior in both the aTAM [1, 17, 23, 24] and 2HAM [4]. On the other hand, the computational and geometric power of the temperature-1 aTAM (and 2HAM) famously remains open [10, 14]).

The difficulty of implementing cooperative binding in experimental DNA tile self-assembly has motivated the study of variants of the temperature-1 aTAM augmented with more practical features that confer similar capabilities. This work has established the temperature-1 aTAM is capable of universal computation and efficient shape assembly if the model is augmented with negative-strength glues [16], a third spatial dimension [7], "triggered" glues [15], or non-square tile shapes [11].

Unfortunately, these variations suffer from a common practical concern: avoiding "spurious nucleation", i.e. binding away from the seed tile. Indeed, even preventing spurious nucleation *with* cooperative binding has substantial challenges [2, 6, 18–20]. The difficulty of implementing aTAM-like seeded growth in experimental systems implies that, at least currently, the behavior of experimental DNA tile self-assembly systems without cooperative binding is captured better by the (temperature-1) 2HAM than the aTAM. Due in part to the newness of the 2HAM, prior study of augmented variants of the temperature-1 2HAM is limited to the staged model [8], a powerful "multi-pot" model.

Our results. Here we obtain three positive results on the computational and geometric behaviors possible in two variants of the temperature-1 2HAM. Since the 2HAM permits growth to begin with any pair of tiles, positive results are necessarily robust for "multiple nucleation" errors.

In the first variant we consider, called the *restricted glue 2HAM* or *rg2HAM*, glue strengths are restricted to -1, 0, or 1. This is the two-handed equivalent of the restricted glue aTAM (rgTAM) [16]. We prove the rg2HAM is Turing universal (Section 3), demonstrating that the "anticooperative" behavior of the rg2HAM, like the cooperative behavior of the aTAM and 2HAM, is capable of simulating any Turing machine computation. The construction critically uses negative-strength glues to "break apart" nucleations encoding incorrect machine computations.

Note that the technique for proving the Turing universality of the temperature-1 3D aTAM [7] cannot be used in the 2HAM, since it uses a long path of tiles that branches and has "incorrect" branches blocked by a previous portion of the path. In the (seedless) 2HAM, an incorrect branch may assemble and become a "junk" assembly. In fact, every temperature-1 3D aTAM construction in [7] has similar problems.

In the second 2HAM variant, the *3D 2HAM* or *3D2HAM*, tiles are (three-dimensional) cubes. This is the two-handed equivalent of the 3D aTAM [7]. We prove that at temperature 1, the 3D2HAM is capable of efficient assembly of $n \times n$ squares (Section 4) and assembly of infinite aperiodic shapes (Section 5), matching known results in the (temperature-1 3D) aTAM [7].

The efficient square construction uses two layers in the third dimension and $O(\log^2 n)$ tile types. The key idea is a special "cyclic" binary counter that prevents incorrect growth from sabotaging completion of the counter's growth.

The aperiodic construction yields only infinite terminal supertiles whose shapes are not translations of themselves, i.e. have no repeating or periodic structure. Prior negative results on assembling aperiodic structures in the temperature-1 aTAM was given as evidence against the Turing universality of that model [10]. Moreover, the two-dimensional temperature-1 2HAM's ability to nucleate growth at any tile in an assembly has recently been shown to imply strong "pumping" results [5, 9] that imply aperiodic systems do not exist.

Here, we contrast these results with the construction of a 3D2HAM system that assembles only infinite aperiodic assemblies. The construction simulates a scaled-up version of a standard aTAM binary counter and special "vacuum" glues to attach infinite periodic rows of this counter to completed (aperiodic) counters, yielding only aperiodic terminal supertiles.

The definition of "aperiodic" used and landscape of results related to aperiodic tile self-assembly systems closely match those of plane tilings: non-overlapping coverings of the plane using collections of shapes called *prototiles* (see [13]). A plane tiling is *aperiodic* provided it has no translational symmetry and a tile set is *aperiodic* provided every plane tiling it admits is aperiodic. Determining whether a prototile set admits a tiling is undecidable; as a corollary, some prototile sets admit only aperiodic tilings [3] (matching [24]). A long-standing conjecture states there are no aperiodic prototile sets from a restricted class of prototile sets, namely singleton tile sets, [12] (matching [17]). This conjecture was recently proved to not be true if a third dimension is allowed [22] (matching [7]).

Turing universality in both tile self-assembly and plane tiling implies the existence of aperiodic instances, but no implications in the other direction are known. Regardless, aperiodic behavior is generally considered evidence for Turing universality. Moreover, such behavior constrains possible proofs of Turing non-universality to those that do not forbid infinite non-repeating behavior.

2 Definitions

The set of *unit vectors* is $U_2 = \{(0,1),(1,0),(0,-1),(-1,0)\}$, also referred to as N, E, S, W, respectively. A *grid graph* is an undirected graph $G = (V, E)$ in which $V \subseteq \mathbb{Z}^2$ and every edge $\{\boldsymbol{a}, \boldsymbol{b}\} \in E$ has the property that $\boldsymbol{a} - \boldsymbol{b} \in U_2$.

Intuitively, a tile type t is a unit square that can be translated, but not rotated, and has a well-defined "side $\boldsymbol{u}$" for each $\boldsymbol{u} \in U_2$. Each side $\boldsymbol{u}$ of t has a *glue* with *label* $\text{label}_t(\boldsymbol{u})$ from some fixed alphabet and a non-negative integer *strength* $\text{str}_t(\boldsymbol{u})$ determined by its type t. Two tiles t and t' that are placed at the points $\boldsymbol{a}$ and $\boldsymbol{a} + \boldsymbol{u}$ respectively, *bind* with *strength* $\text{str}_t(\boldsymbol{u})$ if $(\text{label}_t(\boldsymbol{u}), \text{str}_t(\boldsymbol{u})) = (\text{label}_{t'}(-\boldsymbol{u}), \text{str}_{t'}(-\boldsymbol{u}))$ and with strength 0 otherwise.

2.1 Two-handed Tile Assembly Model

The 2HAM is a generalization of the aTAM where any pair of multi-tile assemblies with sufficient binding strength can attach to each other. Included here is an informal description of the 2D 2HAM; see [4] for a more complete set of definitions.

A *supertile* is the equivalence class of all translations of an assembly, i.e. a "position-less" assembly.[5] The *binding graph* of a supertile is a weighted grid graph whose vertices are tiles and edges between adjacent tiles have weights corresponding to the strength of the binding between them. A supertile is τ-*stable* provided every cut of its binding graph has strength at least τ.

A 2HAM *tile assembly system* (TAS) is a pair $\mathcal{T} = (T, \tau)$, where T is a finite tile set and τ is the *temperature*; typically $\tau \in \{1, 2\}$. Given a TAS $\mathcal{T} = (T, \tau)$, a supertile α is *producible*, denoted $\alpha \in \mathcal{A}[\mathcal{T}]$, provided that either α is a single tile in T, or α is the union of two smaller non-overlapping producible supertiles α_1 and α_2 (called *subassemblies*) such that the cut of α into α_1 and α_2 has strength

[5] Such a distinction is only needed in two-handed models, where the seed cannot be used as a "reference point".

at least τ. For brevity, this relationship between α_1, α_2, and α is (non-uniquely) denoted $\alpha = \alpha_1 + \alpha_2$. A producible supertile α is *terminal*, denoted $\alpha \in \mathcal{A}_{\Box}[\mathcal{T}]$, provided α cannot attach τ-stably to any other producible supertile.

A TAS is *directed* provided it has a unique terminal supertile. Given a connected shape $X \subseteq \mathbb{Z}^2$, we say a TAS $\mathcal{T}$ *self-assembles* X if the shape of every terminal supertile of $\mathcal{T}$ is a translation of X.

2.2 Additional 2HAM definitions

Let α_0 be a producible supertile that grows into β via the supertile assembly sequence $\alpha_0, \alpha_1, \ldots$ and let δ be a producible supertile that can combine with α_0. Then β is *unfair* provided that, for every $i \geq 0$, δ can combine with α_i but does not. Otherwise, we say that β is *fair*. Note that if δ did combine with α, then the resulting supertile does not necessarily grow into β. Intuitively, if a supertile is able to bind to another growing supertile at any given step, it eventually does so if the latter is fair.

A shape is *aperiodic* provided there exists no non-trivial translation of the shape that yields itself. That is, the shape has no translational symmetries. A TAS is *aperiodic* provided every terminal supertile has an infinite aperiodic shape.

2.3 2HAM variants

In the *two-handed restricted-glue Tile Assembly Model* or *rg2HAM*, glue strengths come from the set $\{-1, 0, 1\}$ and there is a unique glue of strength -1. Negative-strength glues permit producible supertiles α and β such that $\gamma = \alpha + \beta$ is *not* τ-stable. Producible supertiles that are not τ-stable can *break* into supertiles along cuts of strength less than τ. A supertile is *terminal* provided it cannot combine with any other producible supertile and cannot break.

In the *3D two-handed Tile Assembly Model* or *3D2HAM*, tiles are unit cubes. As in 2D, the tiles do not rotate, and each face of a tile has a glue.

3 Universal Computation in the rg2HAM

In this section, we prove that Turing-universal computation is possible in the temperature-1 2HAM when a single negative-strength glue is also permitted. Without such a glue, spurious nucleation would seemingly cause the vast majority of produced supertiles to nucleate as encodings of random configurations of the Turing machine and run nonsense computations both forward and backward. This uncountably large sea of undesired supertiles would then dilute supertiles encoding the desired computation.

Our construction, described in the proof of Theorem 1, utilizes small “jackhammer” supertiles that break up the sea of undesired supertiles into constant-sized terminal supertiles (from a constant-sized set). This “jackhammering” is carried out on a “zig-zag” simulation of a modified version of the input Turing machine. Stated formally:

Theorem 1. *Let M be a Turing machine M with tape alphabet Γ. There exists an equivalent machine M'' and tile set T such that for any terminal supertile α of $\mathcal{T} = (T, 1)$ encoding a valid computation of $M''(x)$ and fair $\beta \in \mathcal{A}[\mathcal{T}]$, either β is a subassembly of α or β can be broken apart so that every subassembly of β can grow into a $O(\log |\Gamma|)$-sized terminal supertile.*

To prove Theorem 1, we first define an intermediate machine, M'. Let M' be a Turing machine which is equivalent to M but with the following modifications:

1. A new `end` symbol in the tape alphabet.
2. A new set of `head` symbols in the tape alphabet, one for each element of $\{*\} \times \Gamma$".
3. A new set of tape alphabet symbols that have additional markings to denote that the cells containing them occur to the right of the tape head (with cells to the left unmarked by these symbols). The new symbols are the elements of $\{R\} \times \Gamma$.
4. The assumption that the initial tape contains the input string x padded by one copy of `end` on each side, and with exactly one `head` symbol denoting the location of M's read/write head along with the value of the tape cell.

The tape alphabet of M' has size $3|\Gamma|+1$. M' operates in a "zig-zag" manner, traversing the complete tape from left-to-right, then right-to-left, etc. Each *zig* (left-to-right) or *zag* (right-to-left) traversal carries out at most one step of M, doing so only if the traversal direction matches the direction moved by the head. The left- and rightmost ends of the tape are denoted by the *end* symbol; each time the *end* symbol is reached, the tape is extended by one cell via moving the *end* symbol.

During each zig, M' begins on the leftmost input character and moves right until encountering the `end` symbol. Initially, the leftmost tape cell value contains the `head` symbol and all other tape cells contain the R symbol denoting that they are to the right of the head. The transitions of M' are updated versions of those of M that output values that indicate which side of the head the cell lies on once the head moves. During traversal, the symbols of all cells that do not contain the `head` symbol or are immediately right of such a cell are left unchanged.

For the cell containing a `head` symbol (encoding also the tape symbol in M), if M performs a right-moving transition when in the start state and given the cell value there, then the value of that cell is updated to the output of that transition, the cell to the immediate right is updated to contain a `head` symbol, and the state of M' is updated to encode the new state of M reached after that transition. Cells not containing the head symbol are left unchanged. Identical but symmetric behavior is performed during each zag.

We now define another new Turing machine M'' equivalent to M' but with the following modifications. Let $b = \lceil \texttt{bin}(|\Gamma'|+1) \rceil$, where Γ' is the tape alphabet of M'. That is, b is the length of the binary representation of the size of $|\Gamma'|+1$. Since $|\Gamma'| = 3|\Gamma|+1$, $b = O(\log(|\Gamma|))$. The tape alphabet for M'' is $\{0, 1\}$ and M'' simulates M' by representing each of the characters, or cell values, of M' as a binary string of length b, i.e. each element of Γ' will be assigned a unique b-bit binary string between 1 and $|\Gamma'|$, a new *pad* symbol is represented as $11\dots1$, and the *end* symbol is represented as $00\dots0$.

Also, M'' expects an input tape that encodes (using the previously described encoding) an input tape of M' in binary, but with the *pad* symbol inserted between every pair of cell values and starting with a *pad* symbol on the left side. M'' operates identically to M', but reads, writes, and moves using b steps each, due to encoding each cell of M' as b consecutive cells. As M'' reads a "cell" (b consecutive cells) it writes the *pad* string; M'' then writes the cell's new symbol in the adjoining *pad* string, avoiding simultaneous reading and writing of a cell. Thus after completing each zig or zag, the adjacent cell and *pad* locations alternate, seen in Figure 1. In conclusion, M'' simulates M using a binary tape alphabet and a head that moves in a zig-zag manner.

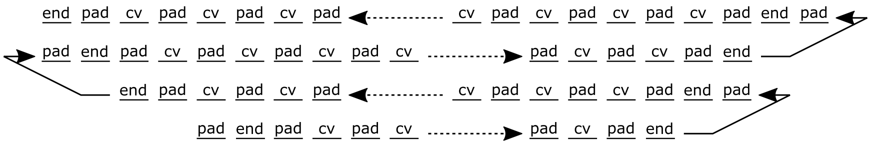
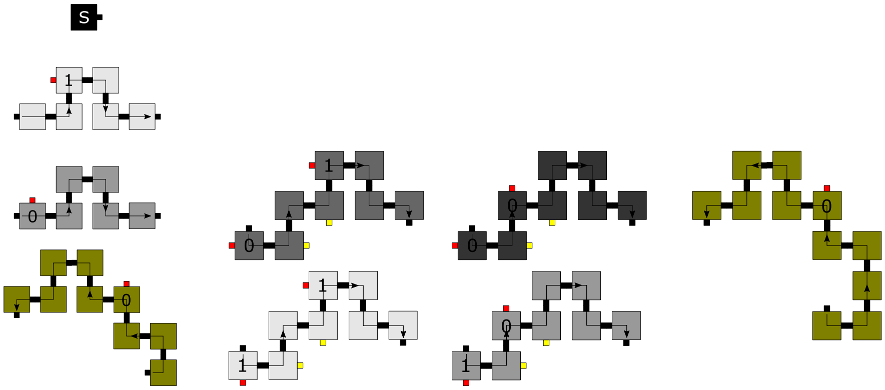

end pad cv pad cv pad cv pad
cv pad cv pad cv pad cv pad end pad
pad end pad cv pad cv pad cv pad cv
pad cv pad cv pad end
end pad cv pad cv pad
cv pad cv pad cv pad end pad
pad end pad cv pad cv
pad cv pad end

Fig. 1: A high-level overview of the growth and tape cell pattern created by $\mathcal{T}$ as it simulates $M(x)$, by simulating the machine M''. The *cv* elements are tape cells and *pad* elements are not.

Next, we create a tile set T such that the rg2HAM system $\mathcal{T} = (T, 1)$ has a unique terminal supertile with size larger than $O(b) = O(\log|\Gamma|)$. This terminal supertile contains an accurate computational history of M'' on the input x' (and thus M on input x) and so simulates $M(x)$. The tile set is the union of small groups of tiles that form functional supertiles called *gadgets*.

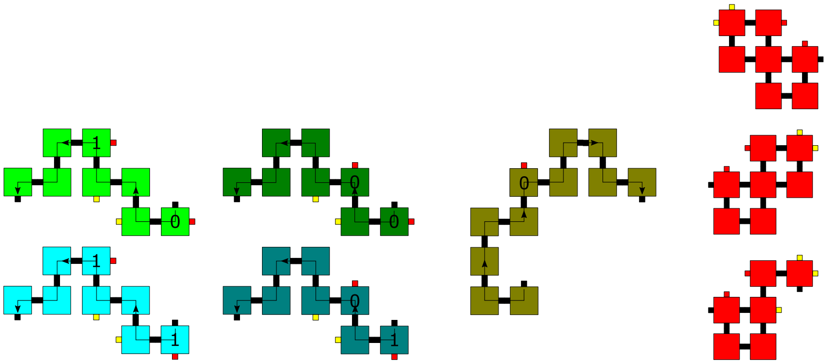
For the analysis of this system, we consider a specific "seeded" growth sequence where tiles attach one at a time, starting with a "seed" that grows into a "seed row" encoding the initial state of the machine, including an input tape. We prove that all other assembly sequences either result in the same terminal supertile (correctly representing the computation and containing the seed), or as "junk" supertiles of size $O(|\log \Gamma|)$.

Figure 2 shows the general structure of gadgets used to form the seed and zig rows. The first set of tiles to be created for T are those that form the seed row. A hard-coded set of gadgets is constructed that only bind to each other and in the correct pattern to encode the seed row: the appropriate binary encoding of x, interspersed by representations of *pad* and with encodings of *end* on the sides). Next, we construct a similarly hard-coded zag row (Fig. 3a) that can only attach to the seed row and is also hard-coded. The next set of gadgets are copies of each of the types of gadgets for zig (Fig. 2b) and zag (Fig. 3a) rows that are specific to each state of M'', so that the current state of M'' is transmitted through the glues and the correct transitions are carried out when at *head* positions.



(a) (b)

Fig. 2: (a) Seed row gadgets. (b) Zig row gadgets. Grey gadgets read bits on the bottom left, and write bits on the top left. Red tabs indicate −1-strength glues; black and yellow tabs indicate 1-strength glues. The gold gadget is the final gadget of a row and does not read but writes a 0 to the new row. Note that for compactness all gold gadgets as depicted write only one 0, but actually write a sequence of b 0's on the right (*end*), then b 1's to the left of those (`pad`), extending the previous row by $2b$ bits. Arrows only show the direction of growth if growth began from the seed.



(a) (b)

Fig. 3: (a) Zag row gadgets. Blue and green gadgets read bits on the bottom right, and write bits on the top right. The gold gadget is the final gadget of a row and does not read but writes a 0 in the new row. Note that arrows only show the direction of growth if growth began from the seed. (b) Jackhammer gadgets. The bottommost is a special type that attaches to a partial seed row that does not contain the seed.

Incorrect growth is “jackhammered” apart by using yellow glues (see Figures 4 and 5) to attach jackhammer and stopper gadgets to attach to the bottoms of rows. These gadgets break apart supertiles with bottom rows that are not the seed row: invalid computations and valid computations spuriously nucleated. Figures 6-8 show an example portion of such a supertile being broken apart into terminal junk supertiles. The resulting broken off pieces have size $O(\log |\Gamma|)$.

Figures 4 and 5 depict example supertiles that do and do not represent valid computations. Figure 6 shows an example of jackhammer gadgets attaching to a supertile that does not represent (part of) a valid computation.

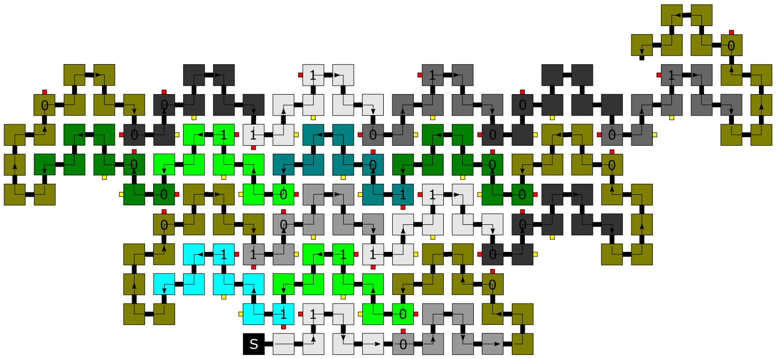


Fig. 4: An example supertile which has correctly grown five rows upward from the hard-coded seed (marked with “S”). The arrows show the direction of growth if growth began from the seed (possible but not necessary).

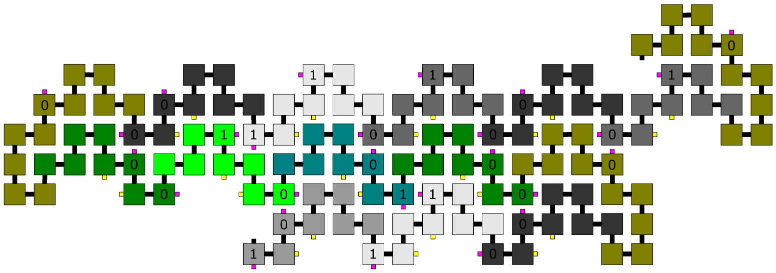


Fig. 5: A supertile which can be assembled but does not contain the hard-coded seed row.

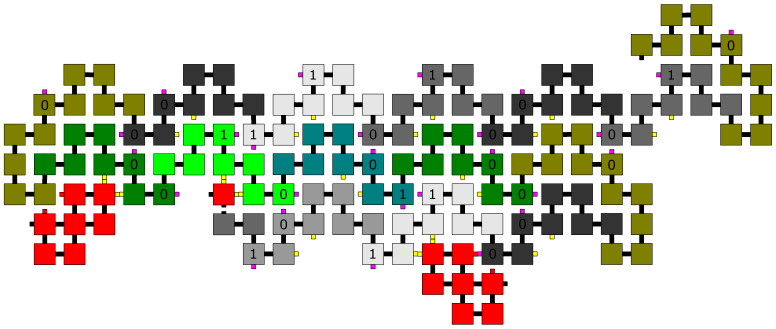


Fig. 6: A few of the locations where (red) jackhammer gadgets can attach. Note that the single-tile partial jackhammer (middle) is currently blocked from further growth. Further breakage by other jackhammers will eventually allow this jackhammer to complete.

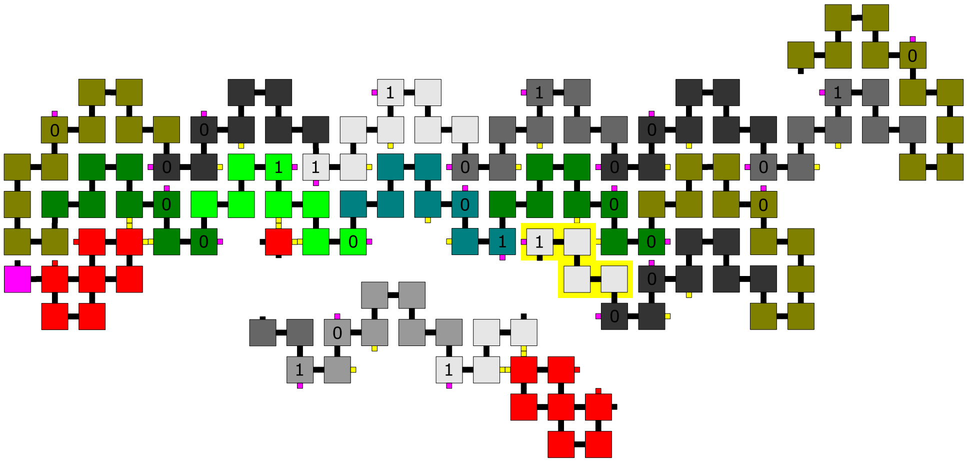


Fig. 7: (Bottom) The minimal supertile which can be separated by the rightmost jackhammer gadget. (Top) The tiles highlighted in yellow show all additional tiles that could have detached with it. Note that now the middle jackhammer is free to grow, and also that the leftmost jackhammer (and any which attach to gadgets at the end of rows) is not able to break off any supertile.

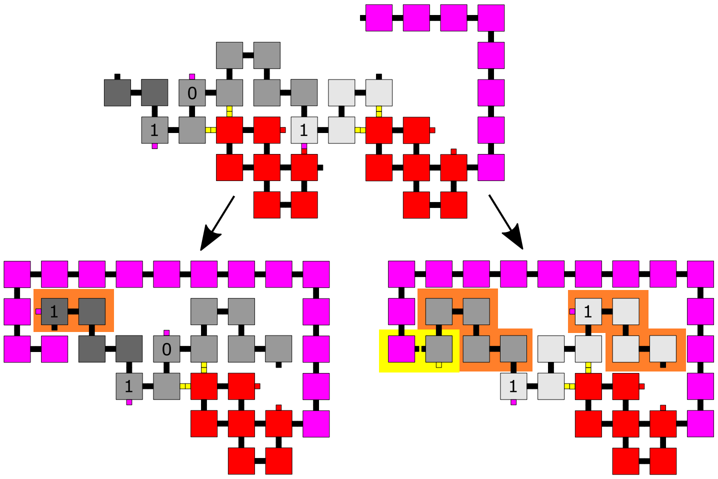


Fig. 8: (Top) The separated supertile from Figure 7 with the attachment of another jackhammer gadget (as well as a portion of a stopper gadget in pink) causing it to break into two pieces. On each of those pieces the stopper gadgets grow. (Left) Orange highlighted portion shows partial regrowth of the row gadget, but the stopper gadget (pink) completes its growth, preventing any more of its growth. (Right) Orange highlighted portions show regrowth of row gadgets, stopper gadget growth is toward potential contention locations (yellow highlighted). If the stopper gadget grows into both of those locations first, the supertile becomes terminal, otherwise further row gadget growth creates a new location where another jackhammer can attach and split the supertile, eventually allowing the stopper gadget to block further growth.

4 Efficient Square Assembly in the Temperature-1 3D2HAM

Here we augment the 2HAM with a third dimension, rather than negative-strength glues, and prove that the additional geometric freedom permits efficient assembly of squares that are one or two layers in the third dimension. More precisely, the construction uses $O(\log^2 n)$ tile types to assemble terminal supertiles that all have a $n \times n \times 2$ bounding box and have a unique $n \times n$ projection into two dimensions. Let $\mathbb{N}_{n-1} = \{0, \ldots, n-1\}$. Formally stated:

Theorem 2. *There exists a tile set T with $|T| = O(\log^2 n)$ such that the 3D2HAM system $\mathcal{T} = (T, 1)$ self-assembles a shape S_n with $\mathbb{N}_{n-1}^2 \times \{0\} \subseteq S_n \subseteq \mathbb{N}_{n-1}^2 \times \{0, 1\}$.*

Proof. The construction begins with the temperature 1 3D aTAM counter of Cook, Fu, and Schweller [7]. This counter simulates the classic temperature-2 zig-zag style counter [17] (see also Section 3 by replacing cooperative binding with geometric blocking.

This counter fails in the the 3D2HAM in two ways. First, all digits of the counter use a common constant-sized set of tiles. In the 3D2HAM, this enables counter rows of unbounded length to grow. Replacing the common tile set with distinct tile sets for each bit of the counter limits rows to exactly the desired quantity of $\Theta(\log n)$ bits.

The second failure is more fundamental: correct growth is only guaranteed when growth starts from the seed and proceeds forward. In the temperature-1 2HAM, assembly can spuriously nucleate at any tile type, allowing the counter to grow backwards from any row. This can lead to erroneous supertiles that increment incorrectly.

As a remedy, we use a *cyclic counter* consisting of two instances of the counter design of Figure 10 that grow in opposite directions and initiate the growth of each other upon completion. See Figure 9 for a schematic of the cyclic counter.

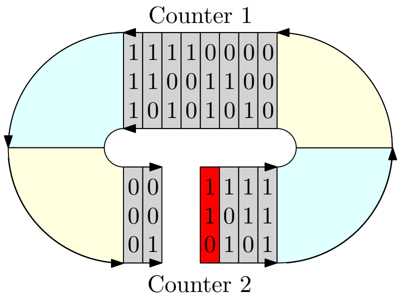


Fig. 9: The cyclic counter combines two forward growth counters that seed each other upon completion. Erroneous backward growth (red) halts, while forward growth proceeds until the crashing into the error to yield a full-length terminal supertile.

Each counter is also modified to halt backwards growth when such growth makes an error. These ideas together resolve the problem of backwards growth, as failed backwards growth is eventually met by (correct) forward growth around the two-counter cycle that necessarily includes a complete correct counter. A fully formed counter is seen in Figure 10. Due to space constraints, the details of how backward growth is halted are omitted.

Assembling $n \times n$ squares. The cyclic counter counts correctly using $O(\log n)$ tile types, but only to values of n that are powers of 2. A counter for a non-power-of-2 value n involves concatenating up to $O(\log n)$ distinct cyclic counters, one for each 1-bit of the binary representation of n. Each counter uses $O(\log n)$ distinct tile types, and the counters are *padded* to a common width. Thus $n \times O(\log n)$ rectangular supertiles for arbitrary n can be assembled using $O(\log^2 n)$ tile types.

Such rectangles are assembled into squares by using one rectangle as a *backbone* and attaching additional rectangles to the backbone at regular $O(\log n)$

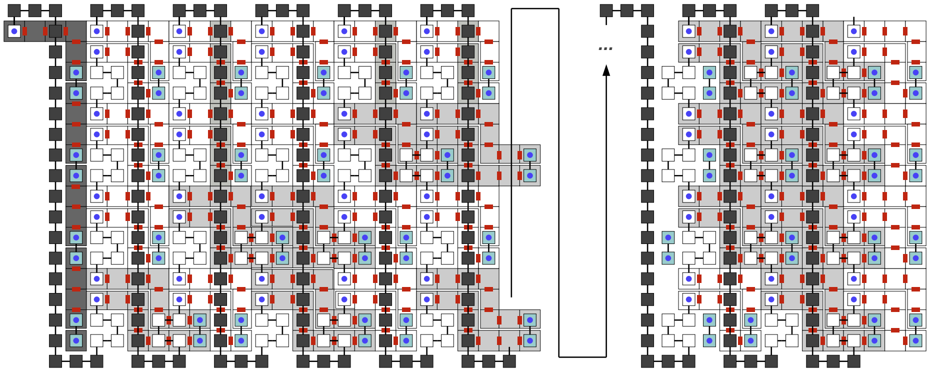


Fig. 10: Forward (left-to-right) growth occurs as in the counter of [7]. The counter value in each column is incremented by reading the geometry of the column to its left. White and grey blocks denote 0 and 1 bits, respectively. Smaller and larger squares denote tiles in the $z = 1$ and $z = 0$ layers, respectively.

intervals. Such spacing can be achieved by modification of the counter to place a special glue based on the values of the least significant $\log \log n$ bits of the counter.

□

5 An Aperiodic 3D $\tau = 1$ 2HAM system

Here we describe an aperiodic system in the three-dimensional 2HAM at temperature 1, complementing the efficient square construction of the previous section.

Theorem 3. *There exists an aperiodic 3D2HAM system* $\mathcal{T} = (T, 1)$.

Proof. Let the aTAM TAS $\mathcal{T}_{\text{count}} = (T_{\text{count}}, \sigma', 2)$ be the well-known system that assembles a "zig-zag" binary counter [17] and let $\alpha_{\text{term}} \in \mathcal{A}_{\Box}[\mathcal{T}_{\text{count}}]$. Our system $\mathcal{T}$ assembles an infinite set of infinite terminal supertiles, each consisting of two reflected scaled versions of α_{term} and some extra "junk".

For every $\alpha \in \mathcal{A}[\mathcal{T}_{\text{count}}]$ there is a supertile in $\mathcal{A}[\mathcal{T}]$ corresponding to α and composed of *macrotiles*: scaled versions of tiles in α where some tile glues are replaced by geometric "dents" and "bumps" encoding the glue's type in binary. Figure 11 shows how glues are replaced by "bit-reading" geometry.

Macrotiles bind to form arbitrarily long *strips* corresponding to binary counter rows in $\mathcal{T}_{\text{count}}$, including *end macrotiles* that match the tiles at the ends of each row. Two strips encoding adjacent binary values can attach vertically via glues on their end macrotiles and matching geometry (see Figure 12).

As in the binary counter of Section 4, the forward growth of the counter is always correct. Thus any assembly in $\mathcal{A}[\mathcal{T}]$ containing the macrotile corresponding to the seed of $\mathcal{T}_{\text{count}}$ corresponds to a subassembly of α_{term}.

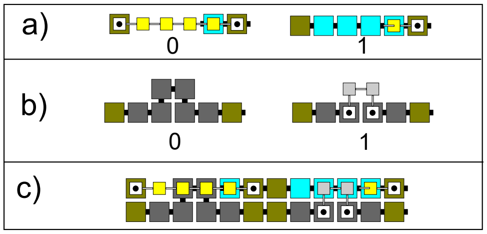


Fig. 11: (a) The bit-readers. Only one of them can assemble after "reading" a bit-writer. The olive-colored tiles attempt to grow both the aqua and yellow tiles but the geometry presented by the bit-writers prevent one of the paths from growing. (b) The two bit-writers. (c) An example of bit-readers "reading" two bit-writers. Smaller and larger squares denote tiles in the $z = 1$ and $z = 0$ layers, respectively.

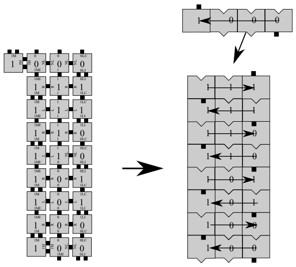


Fig. 12: A producible assembly of $\mathcal{T}_{\text{count}}$ and corresponding macrotile schematic diagram. The geometry of the macrotiles in adjacent strips ensure that adjacent rows encode incremented binary values.

Recall that the aim of $\mathcal{T}$ is to assemble only aperiodic terminal assemblies. Currently, $\mathcal{T}$ has (aperiodic) binary counter and (periodic) counter rows terminal supertiles. Dealing with the infinite periodic rows produced by the (necessarily) unlimited growth is the primary difficulty of the construction. As a solution, two mirrored copies of the counter tile set with identical behavior and disjoint glues are combined and modified to ensure infinite periodic supertiles of one attaching to the aperiodic supertiles of the other. Due to space constraints, we only sketch the implementation of this idea.

Let T_1 be the the tile types described in $\mathcal{T}$ thus far and β_1 be the terminal supertile corresponding to α_{term} assembled from these tile types. Then $\mathcal{T} = (T_1 \cup T_2, 1)$, where T_2 are tile types with glues disjoint from those in T_1 that form a vertically reflected version of β_1 called β_2. Let S_1 be the macrotile in β_1 corresponding to the seed in $\mathcal{T}'_{\text{count}}$ and let S_2 be the counterpart of S_1 in β_2. A `maroon` glue is added to the south face of the southernmost tile of each macrotile in β_1, unless the macrotile corresponds to the seed of $\mathcal{T}'_{\text{count}}$, where a `red` glue is added. Similarly, a `red` glue is added to the north face of the northernmost tile of each macrotile in β_2, unless the macrotile corresponds to the seed of $\mathcal{T}_{\text{count}}$, where a `maroon` is added.

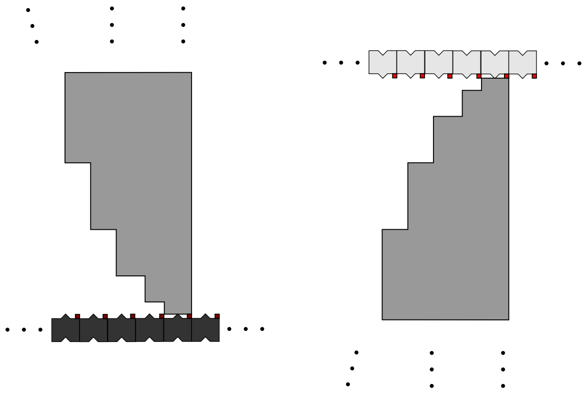


Fig. 13: The moderately shaded regions represent the scaled up macrotile version of the counter supertile and its reflection. The darkly shaded tiles are strips of tiles which expose a `maroon` glue and the lightly shaded tiles are strips of tiles which expose a `red` glue.

Any infinitely long strip is not terminal unless an infinite number of seeds (of the reflected system) bind to their `red` or `maroon` glues (see Figure 13). Moreover, at least one of these seeds must be contained in an infinite counter. Thus every terminal supertile has β_1 or β_2 as a subassembly and so all terminal supertiles of $\mathcal{T}$ are aperiodic. □

References

1. L. Adleman, Q. Cheng, A. Goel, and M.-D. Huang. Running time and program size for self-assembled squares. In *Proceedings of the 33rd Annual ACM Symposium on Theory of Computing (STOC)*, pages 740–748, 2001.
2. R. D. Barish, R. Schulman, P. W. Rothemund, and E. Winfree. An information-bearing seed for nucleating algorithmic self-assembly. *Proceedings of the National Academy of Sciences*, 106(15):6054–6059, March 2009.
3. R. Berger. The undecidability of the domino problem. *Memoirs of the American Mathematical Society*, 66:1–72, 1966.
4. S. Cannon, E. D. Demaine, M. L. Demaine, S. Eisenstat, M. J. Patitz, R. Schweller, S. M. Summers, and A. Winslow. Two hands are better than one (up to constant factors): Self-assembly in the 2HAM vs. aTAM. In *Proceedings of 30th International Symposium on Theoretical Aspects of Computer Science (STACS)*, volume 20 of *LIPIcs*, pages 172–184. Schloss Dagstuhl, 2013.
5. H.-L. Chen, D. Doty, J. Manuch, A. Rafiey, and L. Stacho. Pattern Overlap Implies Runaway Growth in Hierarchical Tile Systems. In L. Arge and J. Pach, editors, *31st International Symposium on Computational Geometry (SoCG)*, volume 34 of *LIPIcs*, pages 360–373. Schloss Dagstuhl, 2015.

6. H.-L. Chen, R. Schulman, A. Goel, and E. Winfree. Reducing facet nucleation during algorithmic self-assembly. *Nano Letters*, 7(9):2913–2919, September 2007.
7. M. Cook, Y. Fu, and R. T. Schweller. Temperature 1 self-assembly: Deterministic assembly in 3D and probabilistic assembly in 2D. In *Proc. of the 22nd ACM-SIAM Sym. on Discrete Algorithms*, SODA'11, pages 570–589, 2011.
8. E. D. Demaine, M. L. Demaine, S. P. Fekete, M. Ishaque, E. Rafalin, R. T. Schweller, and D. L. Souvaine. Staged self-assembly: nanomanufacture of arbitrary shapes with $O(1)$ glues. *Natural Computing*, 7(3):347–370, 2008.
9. D. Doty. Producibility in hierarchical self-assembly. *Natural Computing*, 15(1):41–49, 2016.
10. D. Doty, M. J. Patitz, and S. M. Summers. Limitations of self-assembly at temperature 1. *Theoretical Computer Science*, 412:145–158, 2011.
11. S. P. Fekete, J. Hendricks, M. J. Patitz, T. A. Rogers, and R. T. Schweller. Universal computation with arbitrary polyomino tiles in non-cooperative self-assembly. In *Proc. of the 25th ACM-SIAM Sym. on Discrete Algorithms*, SODA'15, pages 148–167. SIAM, 2015.
12. C. Goodman-Strauss. Open questions in tiling. Online, 2000. http://comp.uark.edu/ strauss/papers/survey.pdf.
13. B. Grünbaum and G. C. Shephard. *Tilings and Patterns.* W.H. Freeman and Company, 1987.
14. P.-E. Meunier, M. J. Patitz, S. M. Summers, G. Theyssier, and D. Woods. Intrinsic universality in tile self-assembly requires cooperation. In *Proceedings of the 25th Symposium on Discrete Algorithms (SODA)*, pages 752–771, 2014.
15. J. E. Padilla, M. J. Patitz, R. Pena, R. T. Schweller, N. C. Seeman, R. Sheline, S. M. Summers, and X. Zhong. Asynchronous signal passing for tile self-assembly: Fuel efficient computation and efficient assembly of shapes. *International Journal of Foundations of Computer Science, Special Issue for UCNC 2013 Full Papers*, 25:459, 2014.
16. M. J. Patitz, R. T. Schweller, and S. M. Summers. Exact shapes and turing universality at temperature 1 with a single negative glue. In *DNA Computing and Molecular Programming*, volume 6937 of *LNCS*, pages 175–189. Springer, 2011.
17. P. W. K. Rothemund and E. Winfree. The program-size complexity of self-assembled squares (extended abstract). In *Proceedings of the 32nd ACM Symposium on Theory of Computing (STOC)*, pages 459–468, 2000.
18. R. Schulman. *The self-replication and evolution of DNA crystals.* PhD thesis, 2007.
19. R. Schulman and E. Winfree. Synthesis of crystals with a programmable kinetic barrier to nucleation. *Proceedings of the National Academy of Sciences*, 104(39):15236–15241, 2007.
20. R. Schulman and E. Winfree. Programmable control of nucleation for algorithmic self-assembly. *SIAM Journal on Computing*, 39(4):1581–1616, 2009.
21. N. C. Seeman. Nucleic-acid junctions and lattices. *Journal of Theoretical Biology*, 99:237–247, 1982.
22. J. E. S. Socolar and J. M. Taylor. An aperiodic hexagonal tile. *Journal of Combinatorial Theory, Series A*, 118(8):2207–2231.
23. D. Soloveichik and E. Winfree. Complexity of self-assembled shapes. *SIAM Journal on Computing*, 36(6):1544–1569, 2007.
24. E. Winfree. *Algorithmic Self-Assembly of DNA.* PhD thesis, Caltech, 1998.