Strong Fault-Tolerance for Self-Assembly with Fuzzy Temperature

David Doty
Department of Computer Science
University of Western Ontario
London, ON, N6A 5B7, Canada
ddoty@csd.uwo.ca

Matthew J. Patitz
Department of Computer Science
University of Texas – Pan American
Edinburg, TX, 78539, USA
mpatitz@cs.panam.edu

Dustin Reishus
Department of Computer Science
University of Southern California
Los Angeles, CA, 90089, USA
reishus@usc.edu

Robert T. Schweller
Department of Computer Science
University of Texas – Pan American
Edinburg, TX, 78539, USA
schwellerr@cs.panam.edu

Scott M. Summers
Department of Computer Science
University of Wisconsin – Platteville
Platteville, WI, 53818, USA
summerss@uwplatt.edu

***Abstract*—We consider the problem of fault-tolerance in nanoscale algorithmic self-assembly. We employ a standard variant of Winfree's abstract Tile Assembly Model (aTAM), the *two-handed* aTAM, in which square "tiles" – a model of molecules constructed from DNA for the purpose of engineering self-assembled nanostructures – aggregate according to specific binding sites of varying strengths, and in which large aggregations of tiles may attach to each other, in contrast to the *seeded* aTAM, in which tiles aggregate one at a time to a single specially-designated "seed" assembly. We focus on a major cause of errors in tile-based self-assembly: that of unintended growth due to "weak" strength-1 bonds, which if allowed to persist, may be stabilized by subsequent attachment of neighboring tiles in the sense that at least energy 2 is now required to break apart the resulting assembly; i.e., the errant assembly is *stable at temperature 2*.**

We study a common self-assembly benchmark problem, that of assembling an $n \times n$ square using $O(\log n)$ unique tile types, under the two-handed model of self-assembly. Our main result achieves a much stronger notion of fault-tolerance than those achieved previously. *Arbitrary* strength-1 growth is allowed; however, any assembly that grows sufficiently to become stable at temperature 2 is guaranteed to assemble into the correct final assembly of an $n \times n$ square. In other words, errors due to insufficient attachment, which is the cause of errors studied in earlier papers on fault-tolerance, are prevented *absolutely* in our main construction, rather than only with high probability and for sufficiently small structures, as in previous fault-tolerance studies.

I. Introduction

Tile-based self-assembly is a model of "algorithmic crystal growth" in which square "tiles" represent molecules that bind to each other via specific and variable-strength bonds on their four sides, driven by random mixing in solution but constrained by the local binding rules of the tile bonds. Beginning with the experimental work of Seeman in the early 1980s [1], such molecules have been engineered from DNA in the laboratory, and used to create a variety of sophisticated self-assembled structures [2], [3] such as Sierpinski triangles and binary counters. Erik Winfree [4], [5], based on experimental work of Seeman [1], modified Wang's mathematical model of tiling [6] to add a physically plausible mechanism for growth through time. Winfree defined two models of tile-based self-assembly, the abstract Tile Assembly Model (aTAM) and the kinetic Tile Assembly Model (kTAM). In both models, the fundamental components are un-rotatable, but translatable square "tile types" whose sides are labeled with glue "colors" and "strengths." Two tiles that are placed next to each other *interact* if the glue colors on their abutting sides match, and in the aTAM, a tile *binds* to an assembly if it interacts on all sides with total strength at least a certain ambient "temperature," usually taken to be 2. In particular, if a tile has two strength-1 glues, both of them must match the corresponding glues in the assembly in order to remain bound.

In the more kinetically plausible kTAM, tiles may bind even if they interact with strength less than 2, but are assumed to detach at a rate inversely and exponentially proportional to the strength with which they interact. Hence tiles attached with strength 1 detach "quickly", and tiles attached with strength 2 detach "slowly". A tile attached with only strength 1 (a so-called "insufficient attachment") represents a potential error, as its other strength-1 glue may be mismatched with the abutting portion of the assembly, or mismatched with what is eventually intended to be placed at that position. However, since strength-1 attachments are assumed to detach after a short time, an insufficient attachment actualizes into a permanent error only if another tile first binds to secure the faulty tile in place, causing the entire assembly to become stable at temperature 2. That is, by "wandering" temporarily through the space of assemblies producible at temperature 1, we may arrive at an assembly not producible at temperature 2, yet that, once formed, is stable at temperature 2. The development of physical and algorithmic mechanisms for preventing such errors remains a formidable challenge in nanoscale self-assembly.

Stated informally, the kTAM refines the aTAM by endowing it with a mechanism for error (temporary binding of tiles with strength 1) as well as a mechanism for error correction (eventual detachment of tiles, even those bound with strength 2). Indeed, numerous papers have used these two mechanisms for high-probability error correction in the kTAM [5], [7]–[11]. In each of these papers except [5], the same basic principle is used to achieve error correction, known as *proofreading*. If an insufficient attachment results in mismatching glues, this error is "amplified" by forcing further growth to require many other insufficient attachments to stabilize. Since these happen only slowly, the assembly process is slowed down, giving time for the tiles that stabilized the original insufficient attachment to detach, thus correcting the original error. In [5], Winfree also shows how errors are removed through the detaching of tiles, although there is no "error-amplification process"; Winfree shows that by setting the ratio of the forward rate to the reverse rate sufficiently small (thus slowing down the entire assembly process), erroneous tiles will detach with high probability.

We work in a variant of the aTAM known as the *two-handed* aTAM (see, for example, [12]). Winfree's original model, the *seeded* aTAM [4], [5], stipulates that assembly begins from a specially-designated "seed" tile type, and all binding events consist of the attachment of a single tile to the growing assembly that contains the seed. The seed thus serves as a *nucleation point* from which all further growth occurs. In reality, such single-point nucleation is difficult to enforce [13], [14] as tiles with matching glues may attach to each other in solution, even if neither of them is connected to the seed tile. The two-handed aTAM models this sort of growth by dispensing with the idea of a seed, and simply defining an assembly to be producible if 1) it consists of a single tile (base case), or 2) it results from the stable aggregation of two producible assemblies (recursive case).

Not only is the two-handed aTAM a more realistic model in the sense of accounting for unseeded nucleation, it allows us to use the geometry of partially-formed assemblies, rather than relying solely on (error-prone) glue specificity, to enforce binding rules between subassemblies. This phenomenon, geometric blocking that prevents bond formation, is well-studied in chemistry and is known as *steric hindrance* [15, Section 5.11] or, particularly when employed as a design tool for intentional prevention of unwanted binding in synthesized molecules, *steric protection* [16]–[18]. Using the mechanism of steric protection, we are able to achieve a much stronger notion of fault-tolerance than that described in previous error-correction papers. Informally, our model of fault-tolerance, which we term the *fuzzy temperature* model, is as follows (a formal description is given in Section IV). Similarly to the kTAM, we allow strength-1 insufficient attachments to occur. However, we do not model forward or reverse rates of growth as in the kTAM, as there is no need to employ the higher reverse rates of insufficient attachments: any insufficient attachments that lead to an assembly that is stable at temperature 2 *were never errors in the first place*, as such an assembly can always lead to an assembly that was producible with only strength-2 growth. That is, viewed as a modification of the aTAM, we allow the temperature to be "fuzzy", occasionally drifting from 2 down to 1, which allows strength-1 growth for as long as the temperature remains low. However, once the temperature is raised back to 2, thus dissolving any structure that is stable only at temperature 1, the stable assemblies that are left over are all assemblies that are already producible at temperature 2 or that can grow into a temperature-2-producible assembly. Therefore, while insufficient attachments can occur, errors due to insufficient attachments cannot occur, since temperature-2 stabilization of such errors, which our construction prevents, is required for the errors to become permanent.

We focus on the problem of assembling an $n \times n$ square, a common benchmark problem for demonstrating the use of self-assembly techniques (see, for example, [19]). In particular, our main result is the construction of a tile set with $O(\log n)$ unique tile types (which is close to the $\Omega(\log n / \log \log n)$ optimal lower bound [19]) that uniquely assembles into an $n \times n$ square in the two-handed aTAM at temperature 2, and that has the fuzzy-temperature fault-tolerance property described above. In keeping with the "wandering" analogy from the beginning of this section, our construction allows arbitrary wandering in the space of assemblies producible at temperature 1, but funnels all such wandering towards a single unique terminal assembly, or towards the oblivion of destruction at temperature 2.[1]

This paper is organized as follows. Section II gives an informal description of the two-handed aTAM. Section III shows a construction of a non-fault-tolerant counter, to introduce some of the main ideas of the full construction. Section IV defines the fuzzy temperature model of fault-tolerance. Section V describes a high-level overview of the main construction and explains the basic techniques employed. Section VI concludes the paper and states open questions.

II. Informal Description of the Two-Handed Abstract Tile Assembly Model

This section gives a brief informal sketch of the two-handed temperature-2 abstract Tile Assembly Model (aTAM).

A *tile type* is a unit square with four sides, each having a *glue* consisting of a *label* (a finite string) and *strength* (0, 1, or 2). We assume a finite set T of tile types, but an infinite number of copies of each tile type, each copy referred to as a *tile*. A *supertile* (a.k.a., *assembly*) is a positioning of tiles on the integer lattice $\mathbb{Z}^2$. Two adjacent tiles in a supertile *interact* if the glues on their abutting sides are equal and have positive strength. Each supertile induces a *binding graph*, a grid graph whose vertices are tiles, with an edge between two tiles if they interact. The supertile is *τ-stable* if every cut of its binding graph has strength at least τ, where the weight of an edge is the strength of the glue it represents. That is, the supertile is stable if at least energy τ is required to separate the supertile into two parts. A *tile assembly system* (TAS) is a pair $\mathcal{T} = (T, \tau)$, where T is a finite tile set and τ is the *temperature*, usually 1 or 2. Given a TAS $\mathcal{T} = (T, \tau)$, a supertile is *producible* if either it is a single tile from T, or it is the τ-stable result of translating two producible assemblies. A supertile α is *terminal* if for every producible supertile β, α and β cannot be τ-stably attached. A TAS is *directed* (a.k.a., *deterministic*, *confluent*) if it has only one terminal, producible supertile. Given a connected shape $X \subseteq \mathbb{Z}^2$, a TAS $\mathcal{T}$ *produces* X *uniquely* if every producible, terminal supertile places tiles exactly on those positions in X (appropriately translated if necessary).

[1]We emphasize that this is *not* the same as saying that our construction assembles an $n \times n$ square at temperature 1: at temperature 1, many different terminal assemblies can nondeterministically form, most of which are junk. Our construction ensures that when the temperature is raised to 2, all the junk dissolves away, leaving only assemblies that are required to assemble the square, and which could have grown anyway had the temperature remained at 2. In fact, it is an open problem, first stated by Winfree and Rothemund in [19], to uniquely assemble an $n \times n$ square at temperature 1 using fewer than $2n - 1$ tile types (compared to our use of $O(\log n)$ tile types). Rothemund and Winfree conjectured that $2n - 1$ is a strict lower bound for this problem.

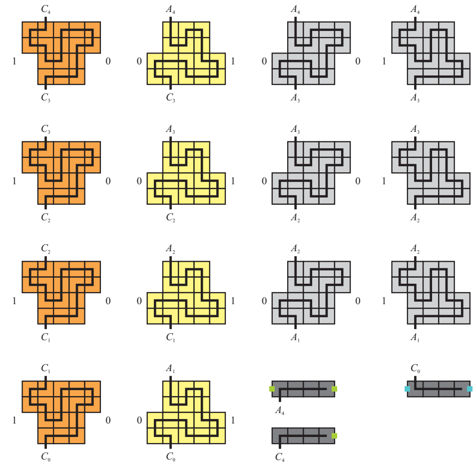


Figure 1: Tile set for two-handed assembly of a length n binary counter using $O(\log n)$ tile types.

III. Two-Handed Assembly of a Counter from $O(\log n)$ Tile Types

In this section we describe the two-handed assembly of a (non-fault-tolerant) counter from $O(\log n)$ tile types, as a warmup to our full fault-tolerant square construction. While this technique does not achieve fault-tolerance, it introduces a novel new counter design technique that utilizes 1) geometry to enforce/restrict specific assemblies and 2) non-determinism of supertile formation and attachment to explore the space of possible intermediate assemblies, despite the existence of only one unique terminal assembly. This technique forms the basis for the more involved fuzzy fault tolerant construction.

The tile set for the counter is depicted in Figure 1. In this figure, an example tile set for a 4 bit counter that counts from 0 to 15 is provided. Tile types that share unique, full strength $\tau = 2$ glues are connected by a black line that crosses over the bonded edge. Other glues in the system include strength $\tau = 2$ glues A_i and C_i for i from 0 to $\log n$ for a length n counter, and two strength $\tau = 1$ glues denoted by the green and blue squares.

Conceptually, the tile set of Figure 1 consists of a number of blocks for each bit position of a binary counter. These blocks assemble into height $O(\log n)$ columns, where the representative block for each bit is determined non-deterministically. Further, the geometry of each block encodes a bit on both the left and right side of the block by a *dent* that appears at either the upper or lower half of the block. In the case of orange *rollover* blocks, the left side encodes the value 1, while the right encodes the value 0; these represent 1 bits less significant than the least significant 0, which all change from 1 to 0 on the next increment. For the yellow *least significant 0* blocks, the left dent encodes the value 0 and the right encodes 1. For the grey *copy* blocks, the left and right encode the same value, with one type of grey block for “1” and another for “0”; these represent bits more significant than the least significant 0, which remain the same on the next increment.

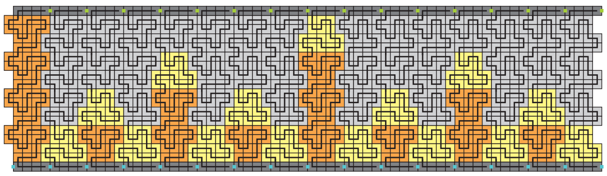


Figure 2: Fully assembled counter from the tiles in Figure 1.

The glue types that connect blocks from one row to another ensure that any assembled column consists of red blocks from rows 1 to r (r at least 1 and at most $\log n$), followed by a yellow block in row $r+1$ (if $r < \log n$), followed by grey blocks (either type) in rows $r+2$ to row $\log n$ (if $r+1 < \log n$). This pattern has the property that for any $(\log n)$-bit string b, a column may assemble that encodes that string in the geometry of the dents on the left side of the column, and the right side of the column in turn encodes $b+1$. Additionally, a fully assembled column can also attach the two four-tile chains of Figure 1 to both the top A glue and bottom C glue of the column. For any two assembled columns, the strength $\tau = 1$ green and blue glues combined give a strength $\tau = 2$ affinity for any two assembled columns to attach. However, due to the rigid teeth-like geometry of the columns, only sequential columns can get close enough to realize the affinity and assemble under the two-handed assembly model. The unique assembly of the tile set of Figure 1 is shown in Figure 2.

In the example provided, we are specifically considering the special case of a counter that grows to a power of 2 length. More generally, it is possible to assemble only columns that encode values greater or equal to a given initial value, thereby allowing the assembly of a length-n counter for general n. However, we leave these details for the extended fault tolerant version of the construction.

The counter in this section is not fuzzy fault tolerant. In particular, the supertile in Figure 3 is producible at temperature $\tau = 1$ (but not $\tau = 2$ because the two-handed model requires that at most 2 supertiles, both of which are stable at $\tau = 2$, combine in any step), stable at temperature $\tau = 2$, but cannot grow into the correct unique $\tau = 2$ assembled counter of Figure 2.

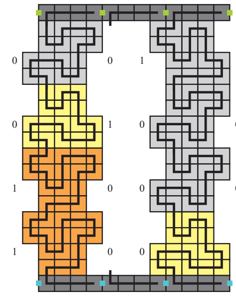


Figure 3: The basic temperature $\tau = 2$ counter in this section is not fuzzy fault tolerant. The above supertile is producible at temperature $\tau = 1$, stable at temperature $\tau = 2$, and cannot grow into the desired unique temperature $\tau = 2$ final assembly of Figure 2.

IV. Fuzzy Temperature Fault-Tolerance

In this section we introduce the fuzzy temperature model of fault-tolerance in self-assembly. The fuzzy temperature assembly model permits rampant temperature $\tau = 1$ growth of supertiles under the two-handed assembly model. We are then interested in what producible temperature $\tau = 1$ assemblies become stable at temperature $\tau = 2$. If even a single temperature $\tau = 1$ assembly becomes stable at temperature $\tau = 2$ and is inconsistent with what can be built in a purely temperature $\tau = 2$ assembly model, the system is deemed *error prone*. On the other hand, if all temperature $\tau = 1$ assemblies that are stable at temperature $\tau = 2$ have a valid temperature $\tau = 2$ path of growth to a supertile that is producible under a pure temperature $\tau = 2$ model, then the system is deemed *fuzzy temperature fault-tolerant*. Put another way, even with arbitrary erroneous strength 1 attachments, a fuzzy temperature fault-tolerant system guarantees that such errors cannot stabilize at temperature 2 unless the stabilized supertile can itself grow into a *correct* temperature $\tau = 2$ assembly, which means such an assembly is not really an error.

Formally, for a given initial tile set T, we define fuzzy temperature fault-tolerance in terms of the following four sets of supertiles: (1) The *dependably produced* (DP) supertiles are those that can be assembled at temperature $\tau = 2$ under the two-handed assembly model. Formally, DP is the set of all producible supertiles for the two-handed assembly system $(T, 2)$; (2) The *dependably terminal* (DT) supertiles are all supertiles in DP that cannot grow any further at temperature $\tau = 2$. Formally, DT is the set of terminal, producible supertiles for the two-handed assembly system $(T, 2)$; (3) The *plausibly produced* (PP) supertiles are those that can be assembled at temperature $\tau = 1$. Formally, PP is the set of all producible supertiles for the two-handed assembly system $(T, 1)$; and (4) The *plausibly stable*

(PS) supertiles are all supertiles in PP that are stable at temperature $\tau = 2$.

Intuitively, DT denotes a final collection of supertiles that can be expected to be built given enough time for assembly in a temperature 2 system. On the other hand, due to the occasional assembly of supertiles with only strength 1 attachments, elements in PP will (plausibly) be assembled. Elements of PP that are not stable at temperature $\tau = 2$ intuitively will eventually break apart and are not of concern. However, these assemblies may grow to a point in which they become stable at temperature $\tau = 2$, in which case they will not break apart. Such assemblies constitute the set PS. The goal is to design a system such that for each element α of PS, every terminal β into which α can grow at temperature $\tau = 2$ is an element of DT (written $PS \Rightarrow DT$), and that DT is the set of desired shapes to be assembled. Put another way, we want to avoid the design of an error prone system in which stable assemblies that are inconsistent with the desired final assembly are built by erroneous $\tau = 1$ strength attachments.

More precisely, the fuzzy temperature fault-tolerance design problem is as follows:

***fuzzy temperature fault-tolerance design problem:**:* Given a target shape Υ, the goal is to design a tile set such that: (1) $PS \Rightarrow DT$ (fuzzy temperature fault-tolerance constraint); and (2) all supertiles in DT have shape Υ. (Desired goal shape is the unique output of the assembly.)

For the remainder of this paper, we attempt to solve the fuzzy temperature fault-tolerance problem for the benchmark example of an $n \times n$ square. As a metric, we are interested in minimizing the number of distinct tile types required to assemble a square while adhering to the fuzzy temperature fault-tolerance constraint; the problem is trivialized if one allows n^2 different tile types to hard-code each position in the square (or even using $O(n)$ tile types to use the non-cooperative "comb" structure from [19]). We show that a sleek $O(\log n)$ tile complexity is achievable, which is very close to the $O\left(\frac{\log n}{\log \log n}\right)$ bound that can be achieved with no fault-tolerance constraint (in the seeded, single-tile addition model).

V. Overview of Fault-Tolerant Square Construction

This section gives a high-level description of the main construction of this paper, a square that assembles under the fuzzy temperature fault tolerance model.

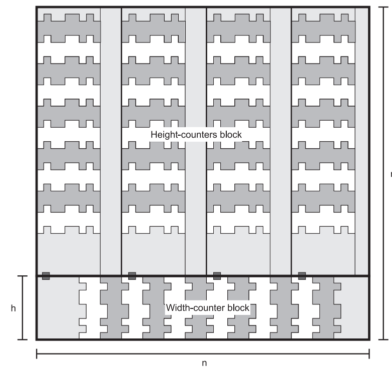


Figure 4: A simplified diagram of the components of a full $n \times n$ square. Components are not represented to scale.

A. Square

As is common in many self-assembly constructions for square-building, most of the work is in constructing counters that calculate the dimensions of the square. Figure 4 shows a high-level diagram of how to compose these counters. The horizontal counter and the vertical counters are constructed in conceptually the same way, with minor differences in the actual implementation. Most of the effort of our main construction is in encoding the number n into the tiles that grow a counter, so that it can control the length to which the counter grows, in a fault-tolerant way.

B. Counter

For simplicity we describe only the horizontal counter. The vertical counters are constructed similarly, with the exception that they are slightly simpler because of the need for the horizontal counter to correctly space out its bonds designed to connect the horizontal counter to the various vertical counters.

Define $k \equiv \lfloor \log n \rfloor + 2$ to be 1 plus the number of bits in n. As in Section III, the counter consists of $\approx n$ columns (actually n divided by the width in tiles of a column, which is a constant, but for simplicity of discussion we will assume that there are n columns), each representing an integer between $2^k - n$ and $2^k - 1$. Note that we refer to columns as "counter-values." Each counter-value is connected to the next by two strength-1 *inter-counter-value glues*, and correct inter-counter-value binding is enforced using bumps and dents as in Section III.

C. Counter-Value

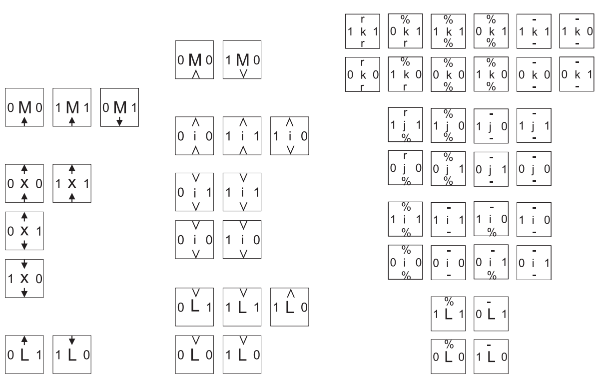
As in Section III, counter-values form randomly from $\approx \log n$ "bit gadgets", each of constant size, with each bit selected at random. Figure 6 shows the bit gadgets, and Figure 7 shows some of them attaching to form a few counter-values of a counter. Beyond the need for fuzzy temperature fault-tolerance, these bit gadgets must meet additional requirements. We first describe how to meet these requirements, and then describe how to achieve fault-tolerance.

1) Glue Design for Additional Requirements of Counter-Values: The logical requirements that counter-values must meet are:

(a) The right side of a counter-value must represent $i+1$ if the left side represents i. This was already needed in Section III.

(b) Each counter-value must be guaranteed to form an integer in the range $\left[2^k - n, 2^k - 1\right]$, so that the counter has exactly n counter-values.

(c) Only a subset of appropriately spaced counter-values should have glues on the north to allow the vertical counters to bind, since the horizontal width of each vertical counters is $\Theta(\log n)$, whereas the horizontal width of each counter-value in the horizontal counter is $O(1)$. This is done by choosing a power of two 2^m (for m just large enough that $2^m >$ width of a vertical counter), and placing the glues to the north every 2^m counter-values.



(a) Increment a binary number

(b) Enforces that the left number is greater than the right

(c) Detects if the binary number formed 1) is equal to a given number, or 2) matches a given number of the low-order bits of that number

Figure 5: Templates for tile sets that perform subsets of the functionality of the hairpin gadgets (described in more detail in Section V-C2). Though not shown, each tile has strength-2 glues on the north and south, implemented in the actual tile set as a pair of single-strength bonds for fault-tolerance purposes. The east and west "bit values" in Figure 5a are represented in the actual tile set by the geometric shape of the 16×16 tile bit gadget that each individual tile in this figure represents, geometrically enforcing agreement on the bits of adjacent bit gadgets.

The fault-tolerance is achieved entirely through the geometric design of the bit gadgets, and the choice of binding paths within them. The requirements (a), (b), and (c) are achieved through careful selection of the north-south glues that connect bit gadgets to each other. For the sake of meeting these three requirements, we can therefore logically view each bit gadget as a single tile, with double-strength glues on the north and south. The values of these glues will then be carried through to every actual tile that makes up a bit gadget, and combined with the glues that hard-code the relative position of each tile in the bit gadget, allow us to conceptually separate the problem of fault tolerance from that of meeting the three requirements discussed above. Finally, we can conceptually separate these three problems from each other, designing tiles to meet those requirements separately, and combine them in a cross-product construction. Figure 5 shows the three tile sets that meet the requirements (a), (b), and (c).

In each case, we take care to ensure that the requirement is met no matter in which order the tiles aggregate. Nonetheless, it is easiest to describe their operation as though the northmost tile is first present, and the counter-value assembles north-to-south; i.e., most significant bit to least significant.

Figure 5a shows the tiles that implement incrementing to ensure that the east bits represent $i+1$ if the west bits represent i. If the position of the least significant 0 in i is p, then all bits at positions above p are equal, all bits at positions below p are 1 for i and 0 for $i+1$, and at position p the bit is 0 for i and 1 for $i+1$. Therefore the tiles nondeterministically guess a position p at which to make this transition, and enforce that all tiles above p have equal bits and all tiles at or below p obey the stated requirement.

Figure 5b shows the tiles that implement range-checking to ensure that the number i that is constructed is greater than $m = 2^k - n$. (Since precisely k bits are assembled, $i < 2^k$.) Imagine comparing i to m starting at the most significant bit. We must enforce that there is at least one bit difference, and that in the position of most significance where there is a difference that the bit from i is 1 and the bit from m is 0. As before, the tiles nondeterministically guess at which position the first disagreement will occur. Below the first disagreement, the bits of i are selected nondeterministically. We chose the value of k so that we know n's most significant bit is 0; this helps to ensure, if tiles grow from south to north and have not yet enforced $i > m$, then the most significant bit of i may be chosen equal to 1 to enforce this.

Figure 5c shows the tiles that ensure that two single-strength glues designed to be an anchor point for vertical counters are placed on the top of a counter-value in the horizontal counter if and only if the counter-value is at an appropriate position to space the vertical counters out evenly. This is accomplished by first determining the number, r, which will be represented by the rightmost counter-value for which this will be the case. Then, whenever the number i represented by a counter-value shares the same least significant m bits with r, the northern glues are present to anchor a vertical counter. Additionally, in the special case where all bits of i match those of r, a pair of northern glues unique to that position are present, to ensure that the special case, rightmost vertical counter with the necessary padding to fill out the width to exactly n, can attach.

2) Geometric Design for Fault-Tolerance: On the assumption that the three requirements in the previous section can be met for each counter-value that forms, we now describe how to use geometry and "synchronization primitives" involving careful placement of glues to ensure that even at temperature 1, unwanted structures cannot grow that will be stable at temperature 2. Recall that at temperature 2, the counter-values of the counter of Section III enforce that binding between adjacent counter-values cannot occur until both counter-values are fully assembled; this occurs because the path (consisting of all strength-2 glues) from one single-strength inter-counter-value glue to another goes through every bump of the counter-value. Hence, to have both glues present, the entire counter-value must also be present.

Our construction enforces that no structure producible even at temperature 1 can stably attach to the east of counter-value i unless it contains enough of the bumps of its westmost counter-value to enforce that binding requires that counter-value to represent $i + 1$. This is enforced by the following constraint: every path (including strength-1 glues) connecting the two inter-counter-value glues of counter-value i that intersects any counter-value $j > i$, also passes through every bump of the counter-value $i + 1$. Therefore, enough of the leftmost counter-value of this structure is guaranteed to be present to ensure that it can only bind to the right of counter-value i if its leftmost counter-value represents $i + 1$.

To enforce that a path from some part of counter-value i to some part of counter-value $i+2$ must traverse the entire height of counter-value $i+1$, we must enforce that a path traverses southward through the bumps of counter-value $i+1$, and then traverses northward again before moving on to counter-value $i + 2$. But since the path cannot "short-circuit" there must be no glues between the southward and northward paths except at the bottom of the counter-value. The bumps and dents on the east side of the southward path must be faithfully represented on the east side of the northward path.

Even though the bits can grow in any order, it is

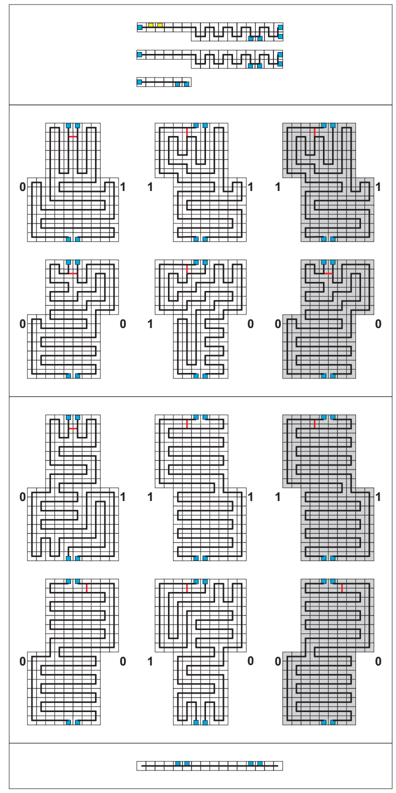


Figure 6: The gadgets that combine to form the counter-values of a counter. The top 6 gadgets that are labeled with bit values are of height 13 rather than 16 for the others, and are used only for the most significant bit in a counter value in order to compensate for the 3 rows of tiles necessary for the gadgets that attach to the top and bottom of the counter and hold the counter values together. Dark black lines represent the strength-2 bonds and forming the bump and dent patterns to represent bit values. The red line is a double bond representing the single point of connection between the two "paths" making up the gadget; see the main text for an explanation of the red bonds' significance. Blue squares represent strength-1 bonds that bind hairpin gadgets to each other and the top/bottom gadgets. Yellow squares represent strength-1 bonds that are used for binding to the vertical counters.

easiest to imagine growing the bits of the southward path, then turning around and guessing those same bits while growing the northward path. Each bit along a single path is represented by what we will call a *hairpin gadget*; one southward and one northward hairpin gadget (though unconnected to each other) form a single bit gadget. To ensure that improper guesses do not result in junk assemblies that cannot grow any further, we use a similar motif to the "single-strength glues at opposite ends" used in Section III, within the hairpin gadgets themselves. That is, hairpin gadgets can only bind stably to the north of other hairpin gadgets when fully formed, which prevents a hairpin gadget that does not match its complementary hairpin gadget from locking in. Figure 7 shows part of a counter formed from these gadgets.

The white hairpin gadgets are "southward growing" (again, if we imagine tracing a path from counter-value i to counter-value $i+1$, bearing in mind that the two-handed assembly can grow in other orders), and the gray hairpin gadgets are "northward growing".

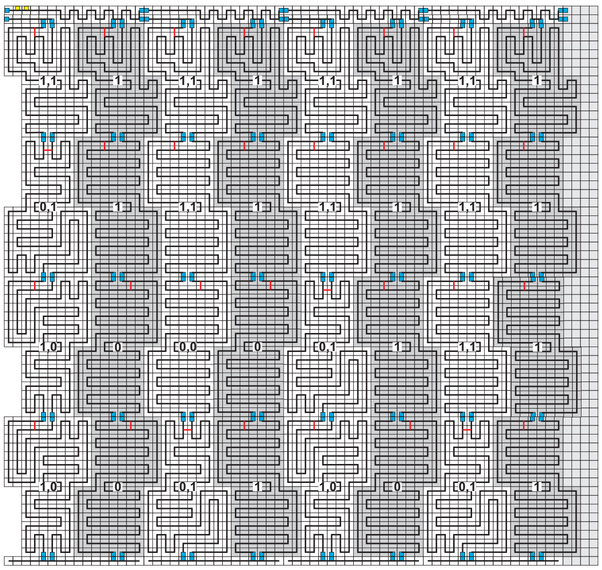


Figure 7: An example of the rightmost 4 columns of the width-counter, counting from 1100 to 1111.

Intuitively, the only connection between a white "left-half" of a counter-value and the gray "right-half" of that counter-value is through the southern row. Northward growth from this row is kept consistent by ensuring that no hairpin gadget can stabilize to the hairpin gadget beneath it until the red double-bond is present. Since every path from this red double-bond to a blue single-bond on the south of the same hairpin gadget goes through the bumps of that gadget, the gadget cannot stabilize unless it is consistent with what has already grown to the left or right of it (and if nothing has already, then *it* determines what must be consistent with it).

Conversely, southward growth, which can lock a hairpin gadget to the hairpin gadget to its north *without* necessarily agreeing with the hairpin to its left or right, nevertheless cannot stabilize at temperature 2 without growing enough of those bumps to enforce agreement. This is because the bottom row must be present to connect a white counter-value half to its gray half, and both must be present to connect that counter-value to the previous (left) counter-value.

VI. Conclusion

Adleman, Cheng, Goel, and Huang [20] show that for each n there is a (seeded, single-tile addition, non-fault-tolerant) tile assembly system that uniquely assembles an $n \times n$ square using $O(\log n / \log\log n)$ unique tile types, a bound that was shown asymptotically tight by Rothemund and Winfree [19]. Since our construction uses $\Theta(\log n)$ tile types, an obvious open question is whether there is a fuzzy temperature fault-tolerant tile assembly system that uses the asymptotically optimal $O(\log n / \log\log n)$ to uniquely assemble an $n \times n$ square. Previous papers [20]–[22] have focused on running time for self-assembled shapes. This is a particularly difficult problem for two-handed assembly. The papers attacking the case of the two-handed model [21], [22] expend much effort to derive the expected assembly time for the much simpler problem of assembling a 1-dimensional $1 \times n$ line from n unique tile types that each encode a different position in the line. It is an open problem, first stated in [21], to prove upper or lower bounds for the optimal time to assemble a square under the two-handed model. It is also an open problem to derive the expected time to completion for our more complicated construction of a fuzzy temperature fault-tolerant square.

The problem of square-building is common in tile-assembly in part because a square is arguably the simplest shape in which the "algorithmic" aspects of self-assembly affect variables such as tile complexity non-trivially, making it a useful benchmark for testing new theoretical techniques. Algorithmic self-assembly deserves the label "algorithmic" because of its computational universality [5]. It is an open problem to show how to simulate an arbitrary algorithm (encoded, for instance, as a Turing machine or a cellular automaton) under the fuzzy-temperature fault tolerance constraint. This would require an appropriate formalization the notion of "simulate an algorithm" under the two-handed aTAM. In the seeded aTAM, for instance, one can state that a *single* tile set T simulates an algorithm A on any input. An input x is given to T by means of arranging some tiles from T into a finite *seed assembly* that represents x in a straightforward way. A possible way (but not necessarily the only way) to solve this open problem would be to show how to construct, for each single-tape Turing machine M and each input string x, a fuzzy-temperature fault tolerant tile system with $O(|M|+|x|)$ tile types that self-assembles an assembly whose rows (each row possibly more than one tile high) represent the entire configuration history of M on input x (which is how a standard simulation in the seeded aTAM proceeds, see for example [19]).

Other proofreading papers such as [8] use a "block-replacement" scheme to convert any tile system (in a certain class such as rectilinear tile systems) into a fault-tolerant tile system, in which each tile in the original tile system is represented by a square block of tiles in the fault-tolerant system. Some tile systems may even have such a conversion done with no scaling factor [9], [23]. It is an open question to show how to do such a conversion on a "natural" class of tile systems, in order to make the resulting tile system fuzzy-temperature fault-tolerant.

Our construction is "floppy": many adjacent tiles in the final square are not connected by glues. One would expect that more strongly connected squares are more physically resilient, and they may also help to enforce the steric protection utilized in our construction, so this floppiness may be a disadvantage. Given the goal of preventing all erroneous temperature-1 growth from stabilizing, it seems unlikely that a *full square* – a square in which every neighboring pair of tiles interact with positive strength – could be constructed using a fuzzy temperature fault-tolerant system. But it is conceivable that more elaborate use of synchronization could allow extra "support substructures" to be used to make our construction "more fully connected", while preserving the fuzzy temperature fault-tolerance. A subproblem is to define a reasonable notion of "floppiness" that is physically meaningful.

Additionally, the two-handed aTAM, while more realistic than the seeded single-tile attachment aTAM in the sense that it allows for nucleation without a seed, is perhaps less realistic in another sense. The DNA tiles that the aTAM was originally conceived to model, while ostensibly two-dimensional, are not necessarily confined to the plane. In particular, the steric protection that we employ requires the tiles in the x-y plane to stay at position $z = 0$. If two mismatching gadgets collide, but one of them "slides" over the other by moving its bumps out of the plane into $z > 0$, then this could allow the cooperative strength-1 bonds to connect even between mismatching gadgets.

However, floppiness is not an unbreakable law of physics; it is an artifact of one particular experimental method of using DNA to create self-assembling tiles. It is not necessarily infeasible to construct tiles by another method that stay in the plane, or thicken them along the z-axis so that some floppiness is tolerable while still enforcing blocking due to steric protection. There are macro-scale techniques for tile self-assembly that are more sturdy and likely to stay in the plane [24], [25], as well as nanoscale techniques for creating rigid DNA structures [26], [27]. It remains an open theoretical problem to design a construction of a fuzzy-temperature fault-tolerant square from $O(\log n)$ tile types that is robust to "3-D floppiness", and an open experimental problem to design physical molecular tiles that are inflexible enough to allow the use of programmed steric protection as a reliable design tool. Another open experimental problem in two-handed tile assembly is to determine, for a given tile implementation, what is the largest size of supertiles that will reliably combine. While it is clear that single tiles experience enough motion in solution to move into positions necessary to combine to growing assemblies, and most likely that supertiles consisting of small numbers of tiles will also do so, there may be an upper bound on the size of supertiles that reliably attach.

Acknowledgment

We thank anonymous referees for suggested improvements to this paper. This research was supported in part by National Science Foundation Grants 0652569 and 0728806 and by a Computing Innovation Fellowship grant to David Doty.

References

[1] N. C. Seeman, "Nucleic-acid junctions and lattices," *Journal of Theoretical Biology*, vol. 99, pp. 237–247, 1982.

[2] P. W. Rothemund, N. Papadakis, and E. Winfree, "Algorithmic self-assembly of DNA Sierpinski triangles," *PLoS Biology*, vol. 2, no. 12, pp. 2041–2053, 2004.

[3] 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*, vol. 106, no. 15, pp. 6054–6059, March 2009. [Online]. Available: http://dx.doi.org/10.1073/pnas.0808736106

[4] E. Winfree, "Algorithmic self-assembly of DNA," Ph.D. dissertation, California Institute of Technology, June 1998.

[5] ——, "Simulations of computing by self-assembly," California Institute of Technology, Tech. Rep. CaltechCSTR:1998.22, 1998.

[6] H. Wang, "Dominoes and the AEA case of the decision problem," in *Proceedings of the Symposium on Mathematical Theory of Automata (New York, 1962)*. Polytechnic Press of Polytechnic Inst. of Brooklyn, Brooklyn, N.Y., 1963, pp. 23–55.

[7] E. Winfree and R. Bekbolatov, "Proofreading tile sets: Error correction for algorithmic self-assembly." in *DNA*, ser. Lecture Notes in Computer Science, J. Chen and J. H. Reif, Eds., vol. 2943. Springer, 2003, pp. 126–144. [Online]. Available: http://dblp.uni-trier.de/db/conf/dna/dna2003.html#WinfreeB03

[8] H.-L. Chen and A. Goel, "Error free self-assembly with error prone tiles," in *Proceedings of the 10th International Meeting on DNA Based Computers*, 2004.

[9] J. Reif, S. Sahu, and P. Yin, "Compact error-resilient computational DNA tiling assemblies," in *DNA: International Workshop on DNA-Based Computers*. LNCS, 2004.

[10] H.-L. Chen, R. Schulman, A. Goel, and E. Winfree, "Reducing facet nucleation during algorithmic self-assembly," *Nano Letters*, vol. 7, no. 9, pp. 2913–2919, September 2007. [Online]. Available: http://dx.doi.org/10.1021/nl070793o

[11] H.-L. Chen, A. Goel, and C. Luhrs, "Dimension augmentation and combinatorial criteria for efficient error-resistant DNA self-assembly," in *Proceedings of the Nineteenth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2008, San Francisco, California, USA, January 20-22, 2008*, S.-H. Teng, Ed. SIAM, 2008, pp. 409–418. [Online]. Available: http://doi.acm.org/10.1145/1347082.1347127

[12] G. Aggarwal, Q. Cheng, M. H. Goldwasser, M.-Y. Kao, P. M. de Espanés, and R. T. Schweller, "Complexities for generalized models of self-assembly," *SIAM Journal on Computing*, vol. 34, pp. 1493–1515, 2005, preliminary version appeared in SODA 2004.

[13] R. Schulman and E. Winfree, "Synthesis of crystals with a programmable kinetic barrier to nucleation," *Proceedings of the National Academy of Sciences*, vol. 104, no. 39, pp. 15 236–15 241, 2007.

[14] ——, "Programmable control of nucleation for algorithmic self-assembly," *SIAM Journal on Computing*, vol. 39, no. 4, pp. 1581–1616, 2009. [Online]. Available: http://link.aip.org/link/?SMJ/39/1581/1

[15] L. G. Wade, *Organic Chemistry*, 2nd ed. Prentice Hall, 1991.

[16] W. Heller and T. L. Pugh, ""Steric protection" of hydrophobic colloidal particles by adsorption of flexible macromolecules," *Journal of Chemical Physics*, vol. 22, no. 10, p. 1778, 1954.

[17] ——, ""Steric" stabilization of colloidal solutions by adsorption of flexible macromolecules," *Journal of Polymer Science*, vol. 47, no. 149, pp. 203–217, 1960.

[18] K. Goto, Y. Hinob, T. Kawashima, M. Kaminagab, E. Yanob, G. Yamamotob, N. Takagic, and S. Nagasec, "Synthesis and crystal structure of a stable S-nitrosothiol bearing a novel steric protection group and of the corresponding S-nitrothiol," *Tetrahedron Letters*, vol. 41, no. 44, pp. 8479–8483, 2000.

[19] P. W. K. Rothemund and E. Winfree, "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*. Portland, Oregon, United States: ACM, 2000, pp. 459–468.

[20] L. Adleman, Q. Cheng, A. Goel, and M.-D. Huang, "Running time and program size for self-assembled squares," in *STOC '01: Proceedings of the thirty-third annual ACM Symposium on Theory of Computing*. Hersonissos, Greece: ACM, 2001, pp. 740–748.

[21] L. Adleman, "Toward a mathematical theory of self-assembly (extended abstract)," University of Southern California, Tech. Rep. 00-722, 2000. [Online]. Available: http://citeseer.ist.psu.edu/272447.html;ftp://ftp.usc.edu/pub/csinfo/tech-reports/papers/00-722.ps.Z

[22] L. Adleman, Q. Cheng, A. Goel, M.-D. Huang, and H. Wasserman, "Linear self-assemblies: Equilibria, entropy and convergence rates," in *In Sixth International Conference on Difference Equations and Applications*. Taylor and Francis, 2001.

[23] D. Soloveichik and E. Winfree, "Complexity of compact proofreading for self-assembled patterns," in *The eleventh International Meeting on DNA Computing*, 2005.

[24] J. Bishop, S. Burden, E. Klavins, R. Kreisberg, W. Malone, N. Napp, and T. Nguyen, "Self-organizing programmable parts," in *Proceedings of the International Conference on Intelligent Robots and Systems*, 2005.

[25] P. W. K. Rothemund, "Using lateral capillary forces to compute by self-assembly," *Proceedings of the National Academy of Sciences*, vol. 97, pp. 984–989, 2000.

[26] D. Liu, M. Wang, Z. Deng, R. Walulu, and C. Mao, "Tensegrity: Construction of rigid DNA triangles with flexible four-arm DNA junctions," *Journal of the American Chemical Society*, vol. 126, pp. 2324–2325, March 2004.

[27] R. P. Goodman, I. A. T. Schaap, C. F. Tardin, C. M. Erben, R. M. Berry, C. F. Schmidt, and A. J. Turberfield, "Rapid chiral assembly of rigid DNA building blocks for molecular nanofabrication," *Science*, vol. 310, no. 5754, pp. 1661–1665, December 2005.