Combining self-healing and proofreading in self-assembly

David Soloveichik · Matthew Cook · Erik Winfree

Abstract Molecular self-assembly is a promising approach to bottom-up fabrication of complex structures. A major impediment to the practical use of self-assembly to create complex structures is the high rate of error under existing experimental conditions. Recent theoretical work on algorithmic self-assembly has shown that under a realistic model of tile addition and detachment, error correcting tile sets are possible that can recover from the attachment of incorrect tiles during the assembly process. An orthogonal type of error correction was recently considered as well: whether damage to a completed structure can be repaired. It was shown that such self-healing tile sets are possible. However, these tile sets are not robust to the incorporation of incorrect tiles. It remained an open question whether it is possible to create tile sets that can simultaneously resist wholesale removal of tiles and the incorporation of incorrect ones. Here we present a method for converting a tile set producing a pattern on the quarter plane into a tile set that makes the same pattern (at a larger scale) but is able to withstand both of these types of errors.

D. Soloveichik (✉)
Department of CNS, California Institute of Technology, Caltech MC 136-93, 1200 E. California Blvd, Pasadena, CA 91126, USA
e-mail: dsolov@caltech.edu

M. Cook
Institute of Neuroinformatics, Zurich, Switzerland
e-mail: cook@ini.phys.ethz.ch

E. Winfree
Department of CNS and CS, California Institute of Technology, Pasadena, CA, USA
e-mail: winfree@caltech.edu

1 Introduction

The Tile Assembly Model (Winfree 1998a, b) formalizes a generalized crystal growth process by which an organized structure can spontaneously form from simple parts. It provides the foundation for theoretically examining how to use self-assembly for massively parallel DNA computation (Winfree 1996; Winfree et al. 1998; Reif 1999; Lagoudakis and LaBean 2000), for creating objects with programmable morphogenesis (Rothemund and Winfree 2000; Adleman et al. 2001; Aggarwal et al. 2005; Soloveichik and Winfree 2007), for patterning of components during nanofabrication of molecular electronic circuits (Cook et al. 2004), and for studying self-replication and Darwinian evolution of information-bearing crystals (Schulman and Winfree 2005a, b). In addition to this theoretical work, several self-assembling systems have been implemented experimentally using DNA molecules as tiles, including both periodic (Winfree et al. 1998; Mao et al. 1999; LaBean et al. 2000) and algorithmic patterns (Mao et al. 2000; Rothemund et al. 2004; Barish et al. 2005).

The Tile Assembly Model considers the growth of two dimensional ''crystals'' made out of square units called tiles. Typically, there are many types of tiles that must compete to bind to the crystal. A new tile can be added to a growing complex if it binds strongly enough. Each of the four sides of a tile has an associated bond type that interacts with abutting sides of other tiles that have already been incorporated. If the two abutting sides have different bond types then their interaction strength is 0. Otherwise, the bond type determines the interaction strength. For tile systems shown in this paper, at least a single strong bond (strength 2) or two weak bonds (strength 1 each) need to be formed for a tile to attach. This is called *error-free* tile addition. The assembly process starts from a specified seed assembly and proceeds by sequential addition of tiles. An *assembly* is an arrangement of tiles that can result by this process. Tiles do not get used up since it is assumed there is an unbounded supply of tiles of each type. If every tile type is ''colored'' a certain way, then the self-assembly process can produce a pattern. Figure 1 illustrates two different patterns and the corresponding tile systems that self-assemble into them. Tile systems, like these, that grow from an L-shaped boundary using only weak bonds are called *quarter-plane* tile systems. Quarter-plane tile systems are a rich class capable of universal computation, and they are the most widely studied type of tile systems in the context of error correction (Winfree 1998b; Winfree and Bekbolatov 2004; Chen and Goel 2005; Reif et al. 2005; Soloveichik and Winfree 2005).

A major stumbling block to making algorithmic self-assembly practical is the error rate inherent in current implementations. While the abstract model supposes that tile additions are error-free and permanent, in reality tile additions are error prone and tiles can dissociate from a growing complex. Further, huge chunks of the structure may be physically ripped off by external mechanical forces, such as shear due to fluid flow during sample handling. Erroneous addition of tiles and wholesale removal of tiles have been examined separately in the literature, so let us review them in turn.

Recent experimental demonstrations of algorithmic self-assembly exhibit error rates of 1–15%: on average every 8th–100th tile that is incorporated does not correctly bond with its neighbors (Rothemund et al. 2004; Barish et al. 2005). Once such a mistake occurs, the erroneous information can be propagated to tiles that are subsequently attached. Thus, a single mistake can result in a drastically different pattern being produced. With this error rate, structures of size larger than roughly 100 tiles cannot be assembled reliably.
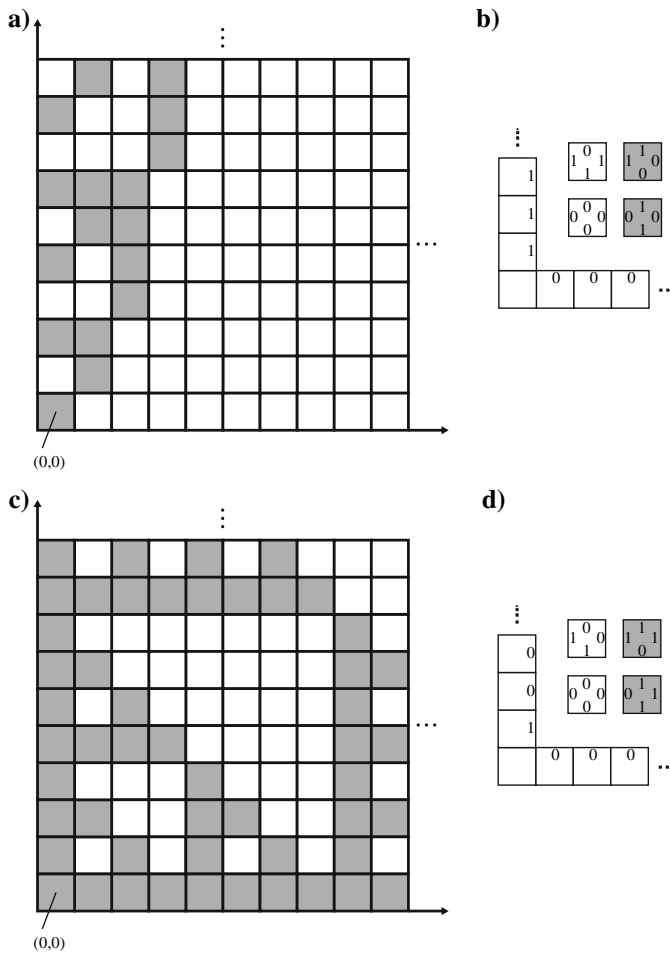
Fig. 1 (**a**) A binary counter pattern and (**b**) a tile system constructing it. (**c**) A Sierpinski pattern and (**d**) a tile system constructing it. The L-shaped boundary (represented in (*a*) and (*b*) as the *x* and *y* axes) is the seed. We assume it is exactly as large as the portion of the pattern we are trying to build. In this formalism, identically-labeled sides match and tiles cannot be rotated. All bond types are weak (strength 1); thus, tiles may attach to the growing assembly only if at least two sides match. Note that the tile choice at each site is deterministic for these two tile sets if the assembly is growing north–east. Growth in the south–west and north–west directions is not deterministic for the counter, and south–west growth is not deterministic for the Sierpinski

The initial work on the relationship between the per tile error rate ε and the rate of assembly r (layers per second) showed that under the best physical conditions $r \propto \varepsilon^2$ for quarter-plane tile systems directly simulating cellular automata (Winfree 1998b). This suggested that the time to produce the $N \times N$ initial portion of the pattern correctly with high probability is $\Omega(N^4)$ (Soloveichik and Winfree 2005). While the physics of the self-assembly process could possibly be modified to achieve a lower probability of the incorporation of incorrect tiles into the growing complex, it is also possible to use some logical properties of the tiles to perform error correction (Winfree and Bekbolatov 2004).
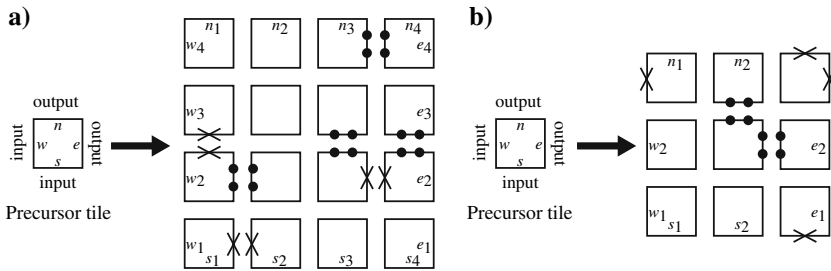
Fig. 2 (**a**) Chen and Goel's snaked proofreading transformations using 4×4 blocks (i.e. $k = 4$), and (**b**) Winfree's self-healing transformations for quarter plane tile systems. Each tile type is replaced with the tile types that fit together to form the block as shown. Strong bonds (strength 2) are indicated with 2 dots. Null bonds (strength 0) bonds are indicated with a cross. All unlabeled (internal) bond types are unique (within the block and between blocks.) The placement of weak and strong bonds is dependent upon the orientation of growth, which is to the north–east for quarter plane tile systems

In this vein, Chen and Goel 2005 developed *snaked proofreading* tile sets that make use of redundant encoding of information to achieve robustness to error (see Fig. 2a). Each tile type of the original quarter-plane tile system is replaced by k^2 tile types that form a block corresponding to the original tile and is colored the same. (The new tile system is no longer quarter-plane since strong (strength 2) and null (strength 0) bonds are used.) If growth occurs without error, the same pattern is produced, albeit at a k times larger scale. However, the pattern of null and strong bonds in a block controls the order of assembly in such a way that an error leads to an assembly whose growth cannot be continued without further errors. Since further errors are unlikely to happen in just the right time and place, growth around erroneous tiles stalls and the erroneous tiles are able to subsequently dissociate, allowing another chance for correct growth. Chen and Goel were able to prove, with respect to a reversible model of algorithmic self-assembly, that error rates decrease exponentially with k, and thus making an $N \times N$ initial portion of the pattern can be done in time $O(N \, poly(\log (N)))$ using only $k = \Omega(\log N)$. Thus theory predicts that using logical error correction we can assemble structures that are correct with high probability significantly faster than the direct implementation of cellular automata.

Extensive damage to the completed parts of the structure was considered in (Winfree 2006). Damage caused by external destructive physical processes is modeled by simply removing some number of tiles from the growing (or completed) structure. Because the assembly model allows crystals to grow in any direction, tiles may begin to fill in holes in the structure from a different direction than the direction of their original growth. While forward growth was deterministic, most of the time backward and sideways growth is not (unless the computation being performed is reversible in some sense). For example, both the binary counter and the Sierpinski pattern do not have deterministic backward growth.[1] *Self-healing* tile sets were developed that perfectly heal such damage to the self-assembled object, assuming that only error-free tile additions occur (see Fig. 2b). Each tile in the original tile set is replaced with 9 tiles as shown in the figure, and thus the pattern is

---

[1] Some backward growth can occur as a result of insufficient attachments in a growing complex that has not been subjected to wholesale tile removal. Chen and Goel's construction handles this type of backward growth but not the more extensive incorrect backward growth possible after tile removal.

produced at a fixed scaleup factor of 3.[2] The construction guarantees that regrowth occurs from the same direction as the original growth by the placement of null bonds that prevent backward growth and strong bonds that allow the assembly process to proceed correctly in the forward direction.

In summary we have two types of errors: (1) tile additions that violate the rule that a tile may only be added if it binds strongly enough, and (2) the removal of tiles despite them being strongly bonded. With existing techniques, each of these types of errors can be controlled separately, but not when they can occur in the same assembly process. Further, simply applying the snaked proofreading transformation followed by the self-healing transformation, or vice versa, does not provide a solution (see the beginning of Sect. 3). In this paper we describe a new construction that has the analogous provable properties as snaked proofreading for the first type of error, but is also able to heal damaged areas where tiles have been removed from the assembly, even when errors in tile addition are allowed.

We assume the reader is familiar with the formal details of the Tile Assembly Model (see (Soloveichik and Winfree 2007) for a long version, and (Soloveichik and Winfree 2005) for a short summary). In the next section we review the model of the dynamics of self-assembly that allows us to speak more precisely about the rate of incorrect tile additions and to show that our construction is robust to such errors. Further, we'll specify more precisely the kind of damage we allow to our assemblies in studying the self-healing property. In the final section, we introduce our construction and prove that it is robust to both types of error. Our proof technique provides an alternative way to Chen and Goel's work of analyzing the error correction process, in that all analysis pertains to individual blocks.

2 Modeling errors

2.1 Erroneous tile additions during growth

To be able to discuss whether or not a tile set is robust to erroneous tile additions, we need a model of the process of incorporation of erroneous tiles into the growing structure. In physical realizations of self-assembly, the growth process involves tiles dynamically attaching and detaching from the assembly. An error occurs if a tile that is held on with total strength less than 2 does not fall off quickly enough and becomes effectively locked in place when another tile attaches such that both tiles are now held on to the rest of the structure with strength at least 2. This event is called an *insufficient attachment*. Thus to determine the effective rate of insufficient attachments we need to study the dynamics of tile attachments and detachments.

Following (Winfree 1998b; Winfree and Bekbolatov 2004; Soloveichik and Winfree 2005) let us define the kinetic Tile Assembly Model (kTAM) as follows. The concentration of each tile type in solution is held constant throughout the self-assembly process, and the concentrations of all tile types are equal. We assume that for every tile association reaction there is a corresponding dissociation reaction. We further assume that the rate of addition (*forward rate f*) of any tile type at any position of the perimeter of the growing assembly is the same. Specifically, $f = k_f e^{-G_{mc}}$ where k_f is a constant that sets the time scale, and G_{mc}

---

[2] Allowing self-assembly to start from a preexisting seed boundary as in this paper, rather than from a single seed tile as in (Winfree 2006), actually permits the use of a simpler transformation that produces a scale-up factor of just 2.

is the logarithm of the concentration of each tile type in solution. The rate that a tile falls off the growing assembly (*reverse rate r_b*) depends exponentially on the number of bonds that must be broken. Specifically, $r_b = k_f e^{-bG_{se}}$ where b is the total interaction strength with which the tile is attached to the assembly, and G_{se} is the unit bond free energy, which may depend, for example, on temperature.[3]

We assume the following concerning f and r_b. As in (Chen and Goel 2005), we let $f = r_2$; then the tile addition requirement imposed by the abstract Tile Assembly Model is satisfied with high probability,[4] forward growth occurs sufficiently quickly[5] and incorrect parts of the assembly can quickly dissociate. In Sect. 4.1 we discuss how close f and r_2 have to be for our proof to work out, but for the purposes of the rest of the paper we assume equality. We assume f (and therefore r_2) can be arbitrarily chosen in our model by changing G_{mc} and G_{se}, for example by changing tile concentrations and temperature. (In practice, there are limits to how much these parameters can be changed.) However, k_f is assumed to be a physical constant not under our control.

Following (Chen and Goel 2005; Soloveichik and Winfree 2005) we make use of a simplification of the kTAM that captures the essential behavior while being more tractable for rigorous proofs. Under the conditions where $f = r_2$, the self-assembly process is dominated by tiles being added with exactly 2 bonds and tiles falling off via exactly 2 bonds. The *locking* kTAM model assumes that these are the only possible single-tile events. That is, $r_b = 0$ for $b \geq 3$ (tiles attached with total interaction strength ≥ 3 never detach), and tiles never attach via a single weak (strength-1) bond. Additionally, insufficient attachments are modeled in the locking kTAM as atomic events, in which two tiles are added simultaneously at any position in which an insufficient attachment can occur. Specifically, any particular pair of tile types that can create an insufficient attachment in the kTAM is added at a rate $f_{err} = O(e^{-3G_{se}})$, which approximates the occurrences of insufficient attachments in the kTAM (Chen and Goel 2005). Thus the total rate of insufficient attachments at a particular location is $Q f_{err}$, where Q is the number of different ways (with different tile types) that an insufficient attachment can occur there. These insufficient attachments are the sole cause of errors during growth.

2.2 Wholesale removal of tiles

Let us now consider how to model the event when (potentially large) portions of the completed pattern are physically ripped off the assembly despite being strongly bonded to it. We simply suppose that any number of tiles can be spontaneously removed from the

---

[3] This formulation ignores the initiation free energy of hybridization, which is non-negligible. See (Winfree 1998b) for details of how this free energy can be treated, yielding a model that is formally identical, but with slightly altered physical meanings for G_{mc} and k_f.

[4] Assuming $f = r_2$, since $r_1 \gg f$, if a tile is added that bonds only with strength 1, it falls off very quickly as it should to obey the aTAM. Tiles attached with strength 2 stick much longer, allowing an opportunity for other tiles to attach to them. Once a tile is bonded with total strength 3, it is very unlikely to dissociate (unless surrounding tiles fall off first).

[5] The approach here and in (Chen and Goel 2005) should be contrasted with the approach in (Winfree 1998b) where the effort is to maximize the "rate of growth" $f - r_2$ while maintaining a low error rate per tile. While for the tile systems considered there, the rate of growth is indeed proportional to $f - r_2$, the tile systems considered here and in (Chen and Goel 2005) can grow quickly even assuming $f = r_2$. This is possible because of strong bonds that bias the assembly forward even if $f = r_2$.

assembly in a distinct event. However, we assume the L-shaped boundary tiles cannot get removed. If the assembled structure becomes disconnected after the event, we assume that the part of the assembly containing the L-shaped boundary remains.

The reason we suppose that the L-shaped boundary cannot get detached is that to make the boundary self-healing requires a different self-healing transformation than the one shown in Fig. 2 (see Winfree 2006), and we wish to keep our argument as simple as possible. It remains an open question whether the self-healing/proofreading construction presented in this paper can be extended to recover the boundary after damage, and whether the techniques used here can be extended to a wider class of tile sets that perform complex growth to create shapes and patterns (Soloveichik and Winfree 2007). (We expect an affirmative answer.)

3 Self-healing proofreading construction

First, let us return to the following issue raised in the Introduction: Why can't we simply apply the snaked proofreading transformation followed by the self-healing transformation, or vice versa, to produce a tile set robust to both insufficient attachments and wholesale removal of tiles? There are two difficulties. The first is of a technical nature: both transformations shown in Fig. 2 only are defined if precursor tiles have weak bonds on all four sides, yet they result in tile sets that also involve both strong and null bonds. Thus the two transformations can't be composed. Sufficiently general (though more complicated) self-healing transformations do exist (Winfree 2006), but although more generally applicable proofreading transformations have been proposed (Winfree and Bekbolatov 2004), there are as yet none with provably good performance. Even supposing this technicality can be overcome, there is no guarantee that the tile set resulting from composing both transformations will retain both robustness properties. One problem is that no matter in which order the transformations are applied, the blocks produced by the last transformation are sensitive to even one insufficient attachment after wholesale removal of tiles. Figure 3 illustrates two incorrect structures that can form and become locked in (according to the locking kTAM). Therefore, we choose to combine the ideas from the snaked proofreading and self-healing constructions, and do not simply compose the transformations directly.
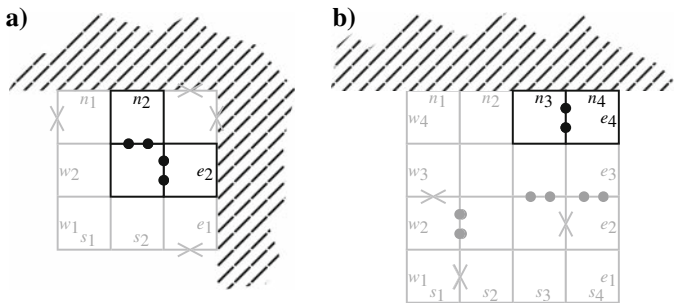

Fig. 3 (**a**) The self-healing and (**b**) the snaked proofreading blocks are sensitive to a few insufficient attachments in the backward growth direction. Consider the case where in the original tile set, backward growth is not deterministic. The structures shown can form after a single insufficient attachment and may be incorrect since they involve backward growth. Every one of the tiles is attached with strength at least 3 and thus cannot dissociate in the locking kTAM. The striped areas show a part of the remaining (correct) assembly after wholesale removal of tiles. The grayed out tiles, which need not be present, show the entire block for reference. There are other structures that can form that cause similar problems

Our self-healing proofreading construction is illustrated in Fig. 4. Intuitively, the construction ensures that erroneous growth (in the correct as well as backward direction) cannot extend far without many insufficient attachments occurring. Specifically, erroneous growths within the S and W rectangles (see Fig. 4b) increase by at most 2 rows or columns after an insufficient attachment. Likewise, the erroneous backward growths bounded by N and E rectangles similarly increase by at most 2 rows or columns after an insufficient attachment. The null bonds (strength 0) on the north–east edges of the block prevent these incorrect growths from meeting up and locking in (i.e. forming a structure where each tile is attached by strength 3 or more). Correct growth, however, can proceed without any insufficient attachments to fill in the south–west quarter of the block, followed by the ''spine'' and the remaining north and east portion of the block. The strong bonds in the spine are required for it to grow. To ensure that both growth of correct tiles and the dissociation of erroneous tiles occurs quickly we rely on series of 1D random walks (see Sect. 4.1). This necessitates the pattern of strong bonds in the rest of the block (outside of the ''spine'').

Suppose we are trying to assemble an $N \times N$ initial portion of the given pattern such that it assembles quickly and correctly with some fixed high probability (like 99%) from the starting L-shaped boundary or from any subassembly that may be formed by removing tiles from the target assembly. We have the following result:[6]

Theorem 1 Fix any constant $\varepsilon > 0$, and consider any N. There is a $k = \Theta(\log N)$ such that using the self-healing proofreading construction with block size k and an L-shaped boundary N blocks long, with an appropriate choice of G_{mc} and G_{se}, the following holds in the locking kTAM model. Starting with any subassembly of A_N containing the L-shaped boundary, with probability at least $1 - \varepsilon$, the initial $N \times N$ block portion of the pattern A_N completes correctly in time $O(N \, poly \, (\log N))$.

As a special case, the subassembly we start out with may just be the L-shaped boundary. Then the assembly process we are talking about is the regular case considered by Chen and Goel which starts out from this boundary. However, we also consider the case where the assembly A_N was damaged by the removal of some tiles. Note that the assumption is that this damage is an isolated event rather than occurring continuously at some rate. If wholesale damage events occur less frequently than the time required to complete the $N \times N$ square, then a reasonable probability of success can be inferred. However, if damage occurs at an intermediate time during assembly—when many incorrect tiles are still present before being replaced by correct growth—then we need a stronger theorem that states that such initial conditions are also allowed. As this requires technical language defining just how much incorrect growth is allowable, we defer this discussion until the end of the proof of the stated theorem.

Proof First we need to make some terms precise. Then we will prove a number of lemmas about the assembly process, and finally with their help we will prove the theorem. In the following, when we say *block* we will mean the square $k \times k$ region which should become filled by a block in the ideal error free assembly. We say a tile in an assembly is *incorrect* if it is not the same tile type as in the ideal error free assembly we are trying to

---

[6] We use the standard asymptotic notation defined as follows: $f(x) = O(g(x))$ means that that there is $c > 0$ such that $f(x) \leq c \cdot g(x)$ for large enough x. Similarly, $f(x) = \Omega(g(x))$ means there is $c > 0$ such that $f(x) \geq c \cdot g(x)$ for large enough x. We write $f(x) = \Theta(g(x))$ if $f(x) = O(g(x))$ and $f(x) = \Omega(g(x))$.
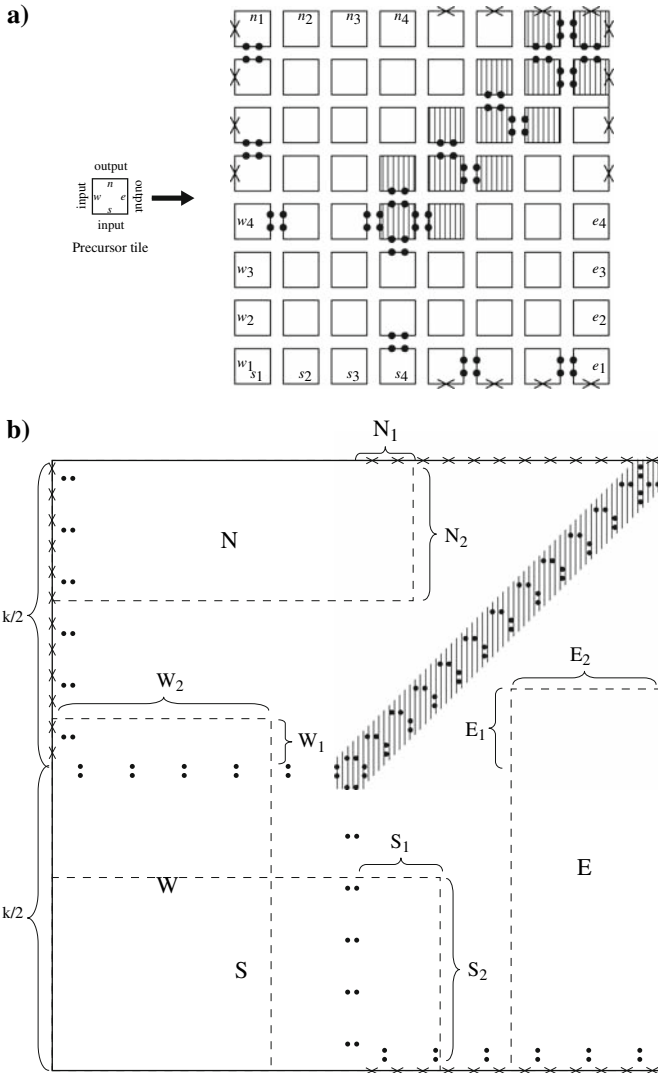
Fig. 4 (**a**) The self-healing proofreading transformations for block size $k = 8$. (**b**) Schematic of the self-healing proofreading block for k divisible by 4. Tiles in the $k \times k$ block are not explicitly drawn; just the pattern of null bonds (X's) and strong bonds (double-dots) are indicated, and all other bonds are weak. For discussing the growth process in the presence of errors, the state of the assembly is characterized by N_i, E_i, S_i, and W_i, which are the smallest non-negative integers such that the following statements hold: Rectangle N (E) contains all incorrect tiles connected to the north (east) side of the block. (We say that a tile is connected to a side of a block if there is a path of connected tiles (abutting with matching bond types) within the block from the given tile to the given side.) Rectangle S (W) contains all incorrect tiles connected to an input side of the block that are incorrect with respect to the west (south) side of the precursor tile. (An incorrect tile must be incorrect with respect to at least one of the input sides of the precursor tile.) The block "spine" consists of tiles shown as striped

produce. Of the directions {north, east, south, west}, we will call the directions west and south the *input* directions, and east and north the *output* directions (because the growth direction is north–east and thus information must pass from the west and south toward

north and east). We say that a block or a region becomes *clean* if all incorrect tiles detach (correct tiles may remain.)

Now refer to Fig. 4b where rectangles N, E, S, W are defined. We define the following in relation to a particular block in a growing assembly:

- State *DOOM*: The block enters this state when an input rectangle W or S touches an output rectangle N or E or if any of the rectangles touches the ''spine'' of the block (marked with striped patterns in the figure). We will see that unless DOOM occurs, all of the rectangles are easy to clean. If DOOM occurs, this can no longer be guaranteed and indeed the block can lock in with incorrect tiles.
- Event *IA*: This event occurs whenever an insufficient attachment happens in the block or its input blocks.
- State *CLEAN*: This state occurs when the block becomes clean, together with the abutting output rectangles of its input blocks. We will demonstrate that after a CLEAN, many IA events are required to enter the DOOM state.
- State *COMPLETE*: The block enters this state when it and its input blocks complete correctly. We will see that once a block enters this state the correct structure is locked in and we can move on to other blocks in our argument.

Lemma 3.2 If a block is COMPLETE then no tile from the block can detach.

Proof By inspection: except for the most south–west tile, every tile in a completed block is attached by strength at least 3. Assuming both input blocks are completed, the most south–west tile is also attached by strength at least 3. □

Lemma 3.3 If a block is CLEAN then (a) at least one IA is needed to get an incorrect tile in the block, (b) at least $k/4$ IA events are needed to enter DOOM, assuming no DOOM occurs in its input blocks first.

Proof Part (a) follows immediately. Unless a DOOM occurs in our block or its input blocks, each insufficient attachment inside our block increases one of N_2, E_2, S_1, S_2, W_1, W_2 by at most 2 or one of N_1, E_1 by at most 1. An insufficient attachment in the west or south input block can increase S_2 or W_2 respectively by 2 (if the incorrect tiles extend into the input blocks).

Thus, the number of IAs must be at least $N_1 + E_1 + (N_2 + E_2 + S_1 + S_2 + W_1 + W_2)/2$. At least one of the following inequalities must hold for DOOM to occur: $N_1 + N_2 \geq k/2 - 1$, $E_1 + E_2 \geq k/2 - 1$, $W_2 \geq k/2 - 1$, $S_2 \geq k/2 - 1$, $W_1 + N_2 \geq k/2$, or $S_1 + E_2 \geq k/2$. Part (b) easily follows. □

Lemma 3.4 The expected time for a block to enter CLEAN is $O(k^3/f)$, assuming (1) no IA occurs, and (2) no DOOM occurs in this block or its input blocks.

Proof Let us first show that the output rectangles of the input blocks become clean in expected time $O(k^3/f)$. Let's consider rectangle E since N can be treated identically. Since no DOOM occurs in this block, we can safely assume that these incorrect tiles are surrounded by either correct tiles or empty space on the north and west, and thus cannot bind to them. Then the largest incorrect structure is illustrated in Fig. 5a. Recall that we are assuming that the forward rate f is equal to the reverse rate r_2 (see Sect. 2.1). Thus the leftmost two rows can fall off via a 1D random walk with expected time $O(k^2/f)$ (see Sect.
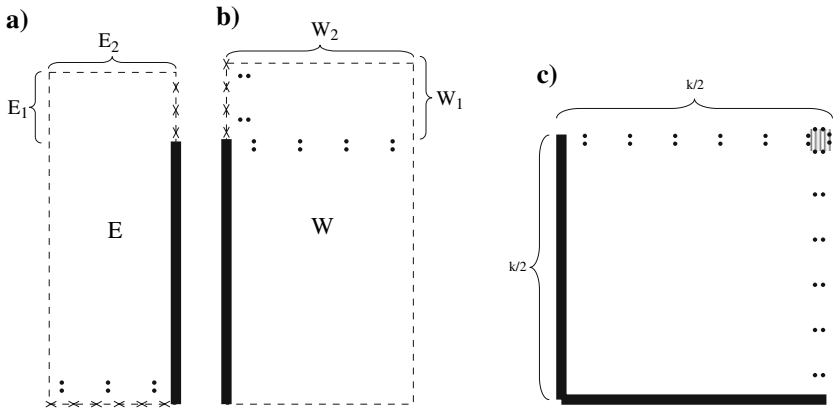
Fig. 5 (**a**, **b**) Illustration for the proof of lemma 3.4. (**c**) Illustration for the proof of lemma 3.5. In (**a–b**) the thick black lines indicate where the incorrect tiles defining the rectangles shown may be bonded to the rest of the assembly (and conversely the dashed lines indicate where they may not be bonded to the rest of the assembly). In (**c**) the thick black lines indicate where the correct tiles are bonded to the rest of the assembly

4.1). Once the two rows fall off, they cannot attach again except via another insufficient attachment. Since there are $O(k)$ pairs of rows, the total expected time for rectangle N to fall off is $O(k^3/f)$.

Once the output rectangles of the input blocks become clean, only correct (or empty) tiles abut our block on the west and south sides. Let's consider the tiles defining rectangle W (rectangle S can be treated similarly). Since these tiles are incorrect with respect to the south side of the block, they cannot be attached to anything on the south side of the rectangle. Further they cannot be attached to anything inside the block along the east or north sides of the rectangle since then those tiles would be part of the rectangle. Since we are assuming that no DOOM occurs, the rectangle cannot extend to and bind the north output border of the block either. Further, the rectangle cannot reach the column of strength-2 bonds on its right because otherwise DOOM would occur (a spine tile would be covered by the rectangle). Thus the rectangle W is as illustrated in Fig. 5b. The $W_2 \times W_1$ top rectangle can falloff via 1D random walks as before. After that, again by the same argument, the rest of rectangle W can fall of in time $O(k^3/f)$. □

Lemma 3.5 The expected time for a block whose input blocks are COMPLETE to enter COMPLETE itself is $O(k^3/f)$, assuming (1) no IA occurs, and (2) no DOOM occurs.

Proof First the block enters CLEAN in expected time $O(k^3/f)$ using Lemma 3.4 (note that no DOOM can occur in the input blocks because they are completed). By Lemma 3.3(a) the block remains free of incorrect tiles. Then let us consider how long it takes to complete the cleaned block whose input blocks are complete, assuming no insufficient attachments occur. Consider the south–west quarter of the block shown in Fig. 5c. Once each row or column completes it is held by strength at least 3 and thus cannot dissociate. Each row or column completes via a 1D random walk with expected time $O(k^2/f)$. Since there are $O(k)$ row/columns, the total expected time to complete the square shown is $O(k^3/f)$. The remaining areas of the block can be completed using a similar argument in time $O(k^3/f)$ as well, after the spine of the block (shown striped in Fig. 4.) completes in time $O(k/f)$. □

Now using these lemmas we can finish the proof of Theorem 3.1. The argument will proceed as follows. First, we set $e^{-G_{se}}$ low enough as a function of the block size k so that insufficient attachments are sufficiently rare that a block has a high probability of entering CLEAN or COMPLETE before an IA occurs. This will ensure that, assuming no DOOM occurs anywhere, the assembly completes in time $O(poly(k) N/\varepsilon)$. Then we will set k large enough as a function of N and ε to ensure that no block enters the DOOM state during the entire assembly process. We will see that k need not be more than logarithmic in N/ε.

Recall that as long as a particular location remains susceptible, we model insufficient attachments at that location as a Poisson process with rate $O(Qe^{-3G_{se}})$ where Q is the number of different tile types that can be added via an insufficient attachment there. Q can be bounded by the total number of different (made up of different tiles) blocks, and since this is the number of tile types in the original tile system, it does not change with k and can be absorbed into the constant. Thus for any block, the distribution of the time until an IA occurs can be upper bounded by an exponential random variable with expected time $t_{ia} = \Omega(1/(k^2 e^{-3G_{se}}))$ since there are no more than 3 k^2 locations where an insufficient attachment can occur (in the block and its input blocks). Let $t_c = O(k^3/f)$ be the worst case expected time for Lemmas 3.4 and 3.5 (over all possible assemblies, blocks and configurations of tiles in blocks). We will want that $t_c \leq$ (1/2) t_{ia}. Recalling that $f = r_2 = O(e^{-2G_{se}})$ (Sect. 2.1), we can guarantee that this inequality is satisfied if we set $e^{-G_{se}}$ low enough: $e^{-G_{se}} = O(1/k^5)$. However, setting $e^{-G_{se}}$ too low slows down the rate of assembly more than necessary, and thus for the following we assume that $e^{-G_{se}} = \Theta(1/k^5)$. Then, $t_c = O(k^{13})$.

We wanted to have $t_c \leq$ (1/2) t_{ia} in order to show that the $N \times N$ blocks are likely, with probability at least $1 - \varepsilon/2$, to assemble correctly in time $O(poly(k) N/\varepsilon)$ assuming no DOOM occurs anywhere. This can be argued as follows. Consider any block whose input blocks are COMPLETE. Lemma 3.5 lets us bound the time until the block itself COMPLETES assuming no IAs or DOOM occur. But what if IAs can occur? The probability that COMPLETE occurs within time $2t_c$ given that an IA doesn't happen in this time is at least 1/2 by Lemma 3.5 and the Markov inequality.[7] The probability that an IA doesn't happen in this time is at least $1/e$ since $2t_c \leq t_{ia}$. Thus the probability that COMPLETE occurs within time $2t_c$ is at least $(1/2)(1/e) = 1/(2e)$. If it doesn't (i.e. an IA occurs or it simply doesn't finish), the probability that a COMPLETE will occur in the next $2t_c$ interval is again at least $1/(2e)$. Thus the expected time until COMPLETE occurs is at most $2e2t_c = 4e\,t_c$. Recall that once a block completes, it can't fall apart (Lemma 3.2). Thus, the current situation is equivalent to irreversible, error-free self-assembly of tiles, where each tile represents a block in our system. In irreversible assembly, the time to assemble an $N \times N$ tile square from an L boundary is on the order of N times the expected time for a single tile to attach (Adleman et al. 2001). Thus, the expected total time to complete the $N \times N$ block assembly is $t_{tot} = O(N \cdot 4\,e\,t_c) = O(N\,t_c)$ assuming no DOOM occurs anywhere. Therefore, the probability that it takes longer than $t_{max} = t_{tot}(2/\varepsilon) = O(N\,k^{13}/\varepsilon)$ to complete the assembly or to get a DOOM somewhere is at most $\varepsilon/2$, again by the Markov inequality.

Next we bound the probability that DOOM occurred anywhere in the assembly in time t_{max}. We'll show that by an appropriate choice of $k = \Theta(\log(N/\varepsilon))$, the probability of this happening is no more than $\varepsilon/2$. Again focus on a particular block; but this time the two input blocks may not be completed. Let us make the worst case assumption that the block remains uncompleted for the duration of assembly t_{max} and thus susceptible to DOOM. We

---

[7] The Markov inequality states that for any non-negative random variable X, $Pr[X \geq a] \leq E[X]/a$ where $E[X]$ is the expected value of X.

want such a block to be without DOOM for the entire time. Recall that the expected time until an IA is bounded by $t_{ia} = \Omega(1/(k^2 e^{-3G_{se}}))$. Thus even with the worst case assumption that the block is never completed, the expected number of IA's for this block in time t_{max} is at most $q = O(t_{max} k^2 e^{-3G_{se}})$. Recalling that $e^{-G_{se}} = O(1/k^5)$, we have $q = O(N/\varepsilon)$. The probability that there will be more than $q N^2$ $(4/\varepsilon)$ is at most $\varepsilon/(4N^2)$ by the Markov inequality. After each IA occurs, with probability at least $1/(2\,e)$ there will be a CLEAN but no IA within time $2\,t_c$ (using the same argument as we followed in the previous paragraph for COMPLETE). Thus with probability at least $1/(2\,e)$, a CLEAN will occur between two IA's. So the probability that among no more than $q N^2$ $(4/\varepsilon)$ IA's, a run of $k/4$ occur in a row without intervening CLEANs can be upper-bounded by $p_{run} = q N^2 (4/\varepsilon)(1 - 1/(2e))^{k/4}$. Since for DOOM to occur, $k/4$ IA's must occur without intervening CLEANs (Lemma 3.3(b)), the probability of DOOM in this block during the entire assembly time is upper bounded by p_{run} if no more than $q N^2$ $(4/\varepsilon)$ IA's occur. If we can somehow guarantee that $p_{run} \leq \varepsilon/(4\,N^2)$, then the probability that DOOM occurs in this block during the entire assembly time t_{max} is at most $\varepsilon/(4\,N^2) + \varepsilon/(4\,N^2) = \varepsilon/(2\,N^2)$. Since there are N^2 blocks, the probability that DOOM will occur even in one of them during the entire assembly time t_{max} is at most $\varepsilon/2$.

Now all that is left is to guarantee that $p_{run} = q\,N^2\,(4/\varepsilon)\,(1-1/(2e))^{k/4} \leq \varepsilon/(4\,N^2)$. Solving for k, we get: $k \geq O(1) \log (16\,N^4\,q/\varepsilon^2) = O(\log (N/\varepsilon))$.

Recall that the total assembly time $t_{max} = O((N/\varepsilon)\,k^{13})$. Using $k = O(\log (N/\varepsilon))$, we get that $t_{max} = O(N{\cdot}poly(\log N))$, for any fixed ε. $\qquad\square$

Note that Theorem 3.1 can be strengthened to allow incorrect tiles in the starting subassembly, as long as there is no DOOM in any of the blocks. Thus we can cover the case in which the assembly process does not entirely complete before the wholesale removal of tiles occurs. However, if this removal occurs periodically and large enough portions of the assembly are taken out each time, it may be that the assembly is never given a chance to complete.

Note also that the dissociation of tiles attached by total strength ≥ 3 is a special case of wholesale tile removal. Thus the proof of the above theorem guarantees that even if the locking assumption is violated, the probability of DOOM in the assembly in time t_{max} is no more than ε. However, we cannot guarantee completion in the sense of Theorem 3.1 since with high likelihood there will be a missing tile somewhere in the assembly even after time t_{max}.

Finally note that DOOM will eventually prevail if we wait long enough.

4 Extensions

4.1 Random walks in the $r_2 \neq f$ regime

Our proofs require that $f = r_2$ exactly. This can't be achieved exactly in practice, which begs the question, is it a necessary requirement for our construction and proof, or can it be relaxed somewhat? We believe it can be only slightly relaxed, due to the competing pressures of needing large patches of incorrect growth to be quickly removed, and at the same time needing correct growth to proceed quickly.

In the proofs of Lemmas 3.4 and 3.5, we model the completion and dissociation of a chain of sequentially added/removed tiles as a continuous time 1D random walk, where the rate with which a tile is added at the end is f and the rate as which the last tile is removed is

r_2. Specifically, we rely on the fact that the expected time for the entire chain to complete (allowing fast forward growth) and the expected time for the chain to fall off (allowing errors to be quickly undone), are both fast (polynomial in the block size k and therefore logarithmic in the size of the target assembly N).

In order to compute the expected time until the entire chain is completed (or equivalently falls off) we can use the following argument. In the discrete time 1D random walk of length $a = O(k)$, the expected number of steps to reach a predetermined end (with the other end being a reflective barrier) is $O(a^2)$ if the forward and backward transition probabilities are equal (Feller 1968). In our case, if $r_2 = f$, the two transition probabilities are indeed equal. Further, since the expected time to take one step is $1/(r_2 + f) = 1/(2f)$, the expected time to reach the predetermined end is $O(a^2/f) = O(k^2/f)$.[8]

However, what happens if the forward rate f does not equal the reverse rate r_2? In the discrete time biased 1D random walk of length a (with a reflecting barrier on the favored end), the expected number of steps to complete in the unfavored direction is $S(\gamma, a) = \frac{1}{2\gamma^2}(1+\gamma)\left[\left(\frac{1+\gamma}{1-\gamma}\right)^a - 1\right] - a/\gamma$ where γ is the difference between the transition probability in the favored and the unfavored directions.[9] This expected time is monotonic increasing in γ and exponentially increasing in a. So if γ is not decreased as we attempt to build larger and larger portions of patterns requiring larger block sizes, then the average number of steps in this random walk grows exponentially with $a = O(k)$, which would not allow us to obtain Theorem 3.1.

Thus, as the block size increases, we need r_2 and f to be closer to each other. As a function of k (and thus ultimately of N) how fast does the difference need to decrease in order for Theorem 3.1 to still hold? Let us assume that G_{se} and thus r_2 is set as required by our proof, but we didn't get G_{mc} quite right such that the actual forward rate f is slightly smaller than r_2. This would normally mean that crystals would be thermodynamically driven to shrink, but since some tile additions form multiple bonds, locking the tile in, assembly is still ratcheted forward. The rate of insufficient attachments can only be smaller and thus still $f_{err} = O(e^{-3G_{se}})$. Thus as long as we can still prove Lemmas 3.4 and 3.5 we would be done. Observe that $\gamma = \frac{r_2 - f}{r_2 + f}$ is the difference between the transition probabilities in the favored and unfavored directions in the corresponding discrete time 1D random walk. Assuming that γ decreases at least as fast as $1/a$, the expected number of steps of the discrete time Markov process to complete in the unfavored direction is no more than $S(\frac{1}{a}, a) = \frac{1}{2}(a^2 + a)\left[\left(\frac{a+1}{a-1}\right)^a - 1\right] - a^2 = O(a^2)$ since $\lim_{a \to \infty}\left(\frac{a+1}{a-1}\right)^a = e^2$. This implies that the expected time for the continuous time Markov process to complete in the unfavored direction is still $O(k^2/f)$ as required for Lemmas 3.4 and 3.5, as long as γ decreases at least as fast as a function in $O(1/k)$.

A thermodynamic argument based on a more realistic kTAM model may require γ to decrease slightly faster, however. In the full kTAM (Winfree 1998b), in which every reaction has a reverse reaction and an equilibrium satisfying detailed balance can be defined, growth of blocks is biased forward if the free energy of adding an entire $k \times k$ block is favorable. This free energy may be calculated as $\Delta n \, G_{mc} - \Delta b \, G_{se}$, where Δn is the number of tiles added, and Δb is the total strength of all new bonds. In our construction, adding a block entails $\Delta n = k^2$ and $\Delta b = 2k^2 + 2$. Thus, favorable growth requires that $\frac{G_{mc}}{G_{se}} < 2 + \frac{2}{k^2}$. Now, since neatly $\gamma = \frac{r_2 - f}{r_2 + f} = \tanh\left(\frac{G_{mc} - 2G_{se}}{2}\right)$, the favorable growth condition

---

[8] At the reflecting barrier the expected time to take a step is twice as large since only the forward direction is possible. However, this does not affect the asymptotic results.

[9] See (Feller 1968) for the general form of the expected duration of 1D discrete time random walks, from which the above expression is derived.

requires that $\gamma < \tanh(G_{se}/k^2)$. Since the proof of Theorem 3.1 required that $e^{-G_{se}} = \Theta(1/k^5)$, $G_{se} = \Theta(\log k)$ and thus the favorable growth condition reduces to $\gamma = O(\tanh(\log k/k^2))$. This is slightly more strict than $\gamma = O(1/k)$ derived in the locking kTAM above.

4.2 Preventing Spurious Nucleation

The blocks produced by our construction have large regions in which tiles are connected to each other via strong (strength 2) bonds (i.e. the ''spine'' of the block, striped pattern in Fig. 4). When the constituent tiles are placed in solution, there is a danger that they will spontaneously nucleate and growth will proceed disconnected from the seed L-shaped boundary. Further, growth may proceed by the aggregation of the separately-nucleated fragments. In other words, our model assumptions that only the assembly containing the L-shaped boundary will grow, and that it will grow by single tile additions, may be violated in practice for such tile sets. Can we avoid large regions of strongly bonded tiles in our construction? We believe ''zig-zag'' boundaries (Schulman and Winfree 2005a) can be adopted to replace the spine, although details remain to be worked out. Rather than a fixed-width spine, this spine would need to be thicker to be more and more robust to spurious nucleation.

4.3 Open problems

Of concern is that an assembly produced by our self-healing proofreading construction has large runs of null bonds along the edges between blocks. In a physical implementation this may result in structural instability and tearing. It is an open question whether self-healing proofreading can be obtained without long runs of null bonds, for example with null bonds more equally spread out throughout the block.

Of further concern is that while Theorem 3.1 shows that asymptotically the block size k scales only logarithmically with the size of the desired pattern, the constants involved may be too large for a physical implementation with current technology. The main obstacle may be the number of tiles types required. For example, the smallest level of protection against insufficient attachments is obtained with $k = 8$ (as in Fig. 4a which requires the introduction of 64 new tile types per tile type of the original non-proofreading system. This may already exceed present capability. However, the constructions presented in this paper are a proof of principle that logical properties of tiles can be used to dramatically reduce the probability of error by simultaneously performing proofreading and self-healing, and it is our hope that future work will discover smaller, more efficient constructions. In fact, while developing the construction presented here, we explored a variety of different approaches that could have yielded quite different constructions. Our choice was guided primarily by the desire for a simple proof, but we believe that there exist a family of constructions that exhibit different trade-offs between effectiveness at proofreading, self-healing, and other characteristics. Further it may be possible to extend these ideas to make tile systems other than quarter plane patterns robust to error.

References

Adleman LM, Cheng Q, Goel A, Huang M-DA (2001) Running time and program size for self-assembled squares. In ACM Symposium on theory of computing (STOC), 740–748

Aggarwal G, Cheng Q, Goldwasser MH, Kao M-Y, de Espanés PM, Schweller RT (2005) Complexities for generalized models of self-assembly. SIAM J Comput 34:1493–1515

Barish RD, Rothemund PWK, Winfree E (2005) Two computational primitives for algorithmic self-assembly: Copying and counting. Nano Lett 5:2586–2592

Chen HL, Goel A (2005) Error free self-assembly using error prone tiles. In: Ferretti C, Mauri G, Zandron C (eds) DNA Computing 10, LNCS vol 3384. Berlin, Springer-Verlag, pp 62–75

Cook M, Rothemund PWK, Winfree E (2004) Self-assembled circuit patterns. In: Chen J, Reif J (eds) DNA Computing 9, LNCS vol 2943. Berlin, Springer-Verlag, pp 91–107

Feller W (1968) An introduction to probability theory and its applications, vol 1. New York, Wiley

Rothemund PWK, Papakakis N, Winfree E (2004) Algorithmic self-assembly of DNA Sierpinski triangles. PLoS Biol 2:e424

Rothemund PWK, Winfree E (2000) The program-size complexity of self-assembled squares. In: ACM symposium on theory of computing (STOC), pp 459–468

LaBean TH, Yan H, Kopatsch J, Liu F, Winfree E, Reif JH, Seeman NC (2000) Construction, analysis, ligation, and self-assembly of DNA triple crossover complexes. J Am Chem Soc 122:1848–1860

Lagoudakis MG, LaBean TH (2000) 2-D DNA self-assembly for satisfiability. In: Winfree E, Gifford DK (eds) DNA Based Computers V, DIMACS vol 54. Providence, RI, American Mathematical Society, pp 141–154

Mao C, LaBean TH, Reif JH, Seeman NC (2000) Logical computation using algorithmic self-assembly of DNA triple-crossover molecules. Nature 407:493–496

Mao C, Sun W, Seeman NC (1999) Designed two-dimensional DNA holliday junction arrays visualized by atomic force microscopy. J Am Chem Soc 121:5437–5443

Reif J (1999) Local parallel biomolecular computing. In: Rubin H, Wood DH (eds) DNA Based Computers III, DIMACS vol 48. Providence, RI, American Mathematical Society, pp 217–254

Reif JH, Sahu S, Yin P (2005) Compact error-resilient computational DNA tiling assemblies. In: Ferretti C, Mauri G, Zandron C (eds) DNA Computing 10, LNCS vol 3384. Berlin, Springer-Verlag, pp 293–307

Schulman R, Winfree E (2005a) Programmable control of nucleation for algorithmic self-assembly. In: Ferretti C, Mauri G, Zandron C (eds) DNA Computing 10, LNCS vol 3384. Berlin, Springer-Verlag, pp 319–328. Extended abstract in DNA Computing 10; preprint of the full paper is cond-mat/0607317 on arXiv.org

Schulman R, Winfree E (2005b) Self-replication and evolution of DNA crystals. In: Capcarrere MS, Freitas AA, Bentley PJ, Johnson CG, Timmis J (eds) Advances in Artificial Life: 8th European Conference (ECAL), LNCS vol 3630. Berlin, Springer-Verlag, pp 734–743

Soloveichik D, Winfree E (2005) Complexity of compact proofreading for self-assembled patterns. In: DNA Computing 11. Berlin, Springer-Verlag

Soloveichik D, Winfree E (2007) Complexity of self-assembled shapes. SIAM J Comput 36:1544–1569

Winfree E (1996) On the computational power of DNA annealing and ligation. In: Lipton RJ, Baum E B (eds) DNA Based Computers, DIMACS vol 27. Providence, RI, American Mathematical Society, pp 199–221

Winfree E (1998a) Algorithmic Self-Assembly of DNA. PhD thesis, California Institute of Technology, Pasadena

Winfree E (1998b) Simulations of computing by self-assembly. Technical Report CS-TR:1998.22, Caltech

Winfree E (2006) Self-healing tile sets. In: Chen J, Jonoska N, Rozenberg G (eds) Nanotechnology: science and computation. Springer-Verlag, Berlin, pp 55–78

Winfree E, Bekbolatov R (2004) Proofreading tile sets: error-correction for algorithmic self-assembly. In: Chen J, Reif J (eds) DNA Computing 9, LNCS vol 2943. Berlin, Springer-Verlag, pp 126–144

Winfree E, Liu F, Wenzler LA, Seeman NC (1998) Design and self-assembly of two dimensional DNA crystals. Nature 394:539–544

Winfree E, Yang X, Seeman NC (1998) Universal computation via self-assembly of DNA: some theory and experiments. In: Landweber LF, Baum EB (eds) DNA Based Computers II, DIMACS vol 44. Providence, RI, American Mathematical Society, pp 191–213