Error Free Self-assembly Using Error Prone Tiles

Ho-Lin Chen[1,⋆] and Ashish Goel[2,⋆⋆]

[1] Department of Computer Science, Stanford University
[2] Department of Management Science and Engineering and (by courtesy) Computer Science, Stanford University, Terman 311, Stanford CA 94305
{holin, ashishg}@stanford.edu

Abstract. DNA self-assembly is emerging as a key paradigm for nanotechnology, nano-computation, and several related disciplines. In nature, DNA self-assembly is often equipped with explicit mechanisms for both error prevention and error correction. For artificial self-assembly, these problems are even more important since we are interested in assembling large systems with great precision.

We present an error-correction scheme, called snaked proof-reading, which can correct both growth and nucleation errors in a self-assembling system. This builds upon an earlier construction of Winfree and Bekbolatov [11], which could correct a limited class of growth errors. Like their construction, our system also replaces each tile in the system by a $k \times k$ block of tiles, and does not require changing the basic tile assembly model proposed by Rothemund and Winfree [8].

We perform a theoretical analysis of our system under fairly general assumptions: tiles can both attach and fall off depending on the thermodynamic rate parameters which also govern the error rate. We prove that with appropriate values of the block size, a seed row of n tiles can be extended into an $n \times n$ square of tiles *without errors* in expected time $\widetilde{O}(n)$, and further, this square remains stable for an expected time of $\widetilde{\Omega}(n)$. This is the first error-correction system for DNA self-assembly that has provably good assembly time (close to linear) and provable error-correction. The assembly time is the same, up to logarithmic factors, as the time for an irreversible, error-free assembly.

We also did a preliminary simulation study of our scheme. In simulations, our scheme performs much better (in terms of error-correction) than the earlier scheme of Winfree and Bekbolatov, and also much better than the unaltered tile system.

Our basic construction (and analysis) applies to all rectilinear tile systems (where growth happens from south to north and west to east). These systems include the Sierpinski tile system, the square-completion tile system, and the block cellular automata for simulating Turing machines. It also applies to counters, a basic primitive in many self-assembly constructions and computations.

⋆ Research supported in part by NSF Award 0323766.
⋆⋆ Research supported by NSF CAREER Award 0339262 and by NSF Award 0323766.

C. Ferretti, G. Mauri and C. Zandron (Eds.): DNA10, LNCS 3384, pp. 62–75, 2005.

1 Introduction

Self-assembly is the ubiquitous process by which objects autonomously assemble into complexes. Nature provides many examples: Atoms react to form molecules. Molecules react to form crystals and supramolecules. Cells sometimes coalesce to form organisms. It is widely believed that self-assembly will ultimately become an important technology, enabling the fabrication of great quantities of small complex objects such as computer circuits. DNA has emerged as an important component to use in *artificial* self-assembly of nano-scale systems due to its small size, its incredible versatility, and the precedent set by the abundant use of DNA self-assembly in nature. Accordingly, DNA self-assembly has received significant attention over the last few years, both by practitioners [13, 15, 10, 11], and by theoreticians [5, 6, 12, 1, 7, 8, 2, 3, 4]. The theoretical results have focused on efficiently assembling structures of a controlled size (the canonical example being assembly of $n \times n$ squares) and shape. In this paper, we are interested in simultaneously achieving robustness and efficiency.

The Tile Assembly Model, originally proposed by Rothemund and Winfree [8], and later extended by Adleman *et al.* [2], provides a useful framework to study the *efficiency* (as opposed to robustness) of DNA self-assembly. In this model, a square tile is the basic unit of an assembly. Each tile has a glue on each side; each glue has a label and a strength (typically 1 or 2). A tile can attach to a position in an existing assembly if at all the edges where this tile "abuts" the assembly, the glues on the tile and the assembly are the same, and the total strength of these glues is at least equal to a system parameter called the temperature (typically 2). Assembly starts from a single seed crystal and proceeds by repeated accretion of single tiles. The speed of an addition (and hence the time for the entire process to complete) is determined by the concentrations of different tiles in the system. Details are in Section 2.

Rothemund and Winfree [8] gave an elegant self-assembling system for constructing squares by self-assembly in this model. Their construction of $n \times n$ squares requires time $\Theta(n \log n)$ and program size $\Theta(\log n)$. Adleman *et al.* [2] presented a new construction for assembling $n \times n$ squares which uses optimal time $\Theta(n)$ and optimal program size $\Theta(\frac{\log n}{\log \log n})$. Both constructions first assemble a roughly $\log n \times n$ rectangle (at temperature 2) by simulating a binary counter, and then complete the rectangle into a square. Later, Adleman *et al.* [3] studied several combinatorial optimization problems related to self-assembly. Together, the above results are a comprehensive treatment of the efficiency of self-assembly, but they do not address robustness.

In nature, DNA self-assembly is often equipped with explicit mechanisms for both error prevention and error correction. For artificial self-assembly, these problems are even more important since we are interested in assembling large systems with great precision. In reality, several effects are observed which lead to a loss of robustness compared to the model. The assembly tends to be reversible, i.e., tiles can fall away from an existing assembly. Also, incorrect tiles sometimes get incorporated and locked into a growing assembly, much like defects in a crystal. However, for sophisticated combinatorial assemblies like counters, which form the basis for controlling the size of a structure, a single error can lead to assemblies drastically larger or smaller (or different in other ways) than the intended structure. Finally, the temperature of the system can be controlled only imperfectly. Experimental studies of algorithmic self-assembly have observed error rates of 1% to 10% [11].

The work towards robustness has focused so far on two broad approaches. The first approach is to identify mechanisms used by nature for error-correction and error-prevention in DNA self-assembly and study how they can be leveraged in an algorithmic setting. One example of this approach is strand invasion [4]. The other approach is to design more combinatorial error-correction mechanisms. This is closest in spirit to the field of coding theory. One example of this approach, due to Winfree and Bekbolatov [11], is proof-reading tiles. They suggest replacing each tile in the original system with a $k \times k$ block. This provides some redundancy in the system (hence the loose analogy with coding theory). Their approach can correct *growth errors*, which result from an incorrect tile attaching at a correct location, i.e., a location where some other tile could have correctly attached. However, their approach does not reliably correct *nucleation errors*, which result from a tile (correct or incorrect) attaching at a site which is not yet active. Their proof-reading scheme is explained in section 3, along with the difference between growth and nucleation errors.

We present a modified proof-reading system which can correct both kinds of errors; we call it a snaked proof-reading system. Our scheme provably (under some mild assumptions) results in error-free assembly of an $n \times n$ square in time $\widetilde{O}(n)$ with high probability (whp). Further, our system results in the final assembly remaining stable for an $\Omega(n)$ duration whp. Hence, there is a large window during which there is a high probability of finding complete assemblies. The best-possible assembly time for an $n \times n$ structure is linear even without errors and even in the irreversible model. Thus, our system guarantees close to optimum speed. To the best of our knowledge, this is the first result which simultaneously achieves both robustness and efficiency.

Our snaked system is explained informally in section 3 using an illustrative example. We prove that the error-rate in this illustrative example is much better for our system than for that of Winfree and Bekbolatov. We give a formal description of our system in section 4 and prove the properties of error-correction and efficiency. Section 4 also provides simulation evidence with both our illustrative example and the Sierpinski tile system [10]; in both cases, we demonstrate that our system resulted in a significant reduction in errors.

Our analysis uses the thermodynamic model of Winfree [10]. We assume that the forward and reverse rates as well as the error-rates are governed by underlying thermodynamic parameters. We first analyze the performance of $k \times k$ proof-reading blocks in terms of the error-rate and efficiency, and then let k grow to $O(\log n)$. Our $\widetilde{O}$ notation hides polynomials in $\log n$. We believe that our analysis is slack, and can be significantly improved in terms of the dependence on k. We make some simplifying assumptions to allow our proofs to go through; our simulations indicate that these assumptions are just an artifact of our analysis and not really necessary.

Our basic construction (and analysis) applies to all rectilinear tile systems (where growth happens from south to north and west to east). These systems include the Sierpinski tile system, the square-completion tile system, and the block cellular automata for simulating Turing machines. It also applies to counters, a basic primitive in many self-assembly constructions and computations, but we omit the discussion about counters from this paper.

2 Tile Assembly Model

2.1 The Combinatorial Tile Assembly Model

The tile assembly model was originally proposed by Rothemund and Winfree[8, 2]. It extends the theoretical model of tiling by Wang [9] to include a mechanism for growth based on the physics of molecular self-assembly. Informally, each tile of an assembly is a square with glues of various types on each edge. Two tiles will stick to each other if they have compatible glues. We will present a succinct definition, with minor modifications for ease of explanation.

A tile is an oriented unit square with the north, east, south and west edges labeled from some alphabet Σ of glues. For each tile t, the labels of its four edges are denoted $\sigma_N(t)$, $\sigma_E(t)$, $\sigma_S(t)$, and $\sigma_W(t)$. Sometimes we will describe a tile t as the quadruple $(\sigma_N(t), \sigma_E(t), \sigma_S(t), \sigma_W(t))$. Consider the triple $< T, g, \tau >$ where T is a finite set of tiles, $\tau \in \mathbf{Z}_{>0}$ is the *temperature*, and g is the *glue strength* function from $\Sigma \times \Sigma$ to $\mathbf{Z}_{\geq \mathbf{0}}$, where Σ is the set of glues. It is assumed that for all $x, y \in \Sigma$, $(x \neq y)$ implies $g(x, y) = 0$ and there's a glue *null* $\in \Sigma$, such that $g(null, x) = 0$ for all $x \in \Sigma$. A *configuration* is a map from $\mathbf{Z}^2$ to $T \bigcup empty$.

Let C and D be two configurations. Suppose there exist some $t \in T$ and some $(x, y) \in \mathbf{Z}^2$ such that $D = C$ except at (x, y), $C(x, y) = null$ and $D(x, y) = t$. Let $f_{N,C,t}(x, y) = g(\sigma_N(t), \sigma_S(C(x, y+1))$. Informally $f_{N,C,t}(x, y)$ is the strength of the bond at the north side of t under configuration C. Define $f_{S,C,t}(x, y)$, $f_{E,C,t}(x, y)$ and $f_{W,C,t}(x, y)$ similarly. Then we say that tile t is *attachable* to C at position (x, y) iff $f_{N,C,t}(x, y) + f_{S,C,t}(x, y) + f_{E,C,t}(x, y) + f_{W,C,t}(x, y) \geq \tau$, and we write $C \rightarrow_{\mathbf{T}} D$ to denote the transition from C to D in attaching a tile to C at position (x, y). Informally, $C \rightarrow_{\mathbf{T}} D$ iff D can be obtained from C by adding a tile t such that the total strength of interaction between t and C is at least τ.

A *tile system* is a quadruple $\mathbf{T} =< T, s, g, \tau >$, where T, g, τ are as above and $s \in T$ is a special tile called the "seed". We define the notion of a *derived supertile* of a tile system $\mathbf{T} =< T, s, g, \tau >$ recursively as follows:

1. The configuration Γ such that $\Gamma(x, y) = \mathbf{empty}$ except when $(x, y) = (0, 0)$ and $\Gamma(0, 0) = s$ is a derived supertile of $\mathbf{T}$, and
2. if $C \rightarrow_{\mathbf{T}} D$ and C is a supertile of $\mathbf{T}$, then D is also a derived supertile of $\mathbf{T}$.

Informally, a derived supertile is either just the seed (condition 1 above), or obtained by legal addition of a single tile to another derived supertile (condition 2). We will often omit the word "derived" in the rest of the paper, and use the terms "seed supertile" or just "seed" or s to denote the special supertile in condition 1.

A *terminal supertile* of the tile system $\mathbf{T}$ is a derived supertile A such that there is no supertile B for which $A \rightarrow_{\mathbf{T}} B$. If there is a terminal supertile A such that for any derived supertile B, $B \rightarrow^*_{\mathbf{T}} A$, we say that the tile system *uniquely produces* A. Given a tile system $\mathbf{T}$ which uniquely produces a supertile, we say that the program size complexity of the system is $|T|$ i.e. the number of tile types.

2.2 The Kinetic Model: Rates and Free Energy

Adleman *et al.* presented a model for running time of reversible self-assemblies [2]. In this paper, we use a kinetic model proposed by Winfree which computes the forward and reversed rate as functions of thermodynamic parameters [10]. It has the following assumptions:

1. Tile concentrations are held constant throughout the self-assembly process.
2. Supertiles do not interact with each other. The only two reactions allowed are addition of a tile to a supertile, and the dissociation of a tile from a supertile.
3. The forward rate constants for all tiles are identical.
4. The reverse rate depends exponentially on the number of base-pair bonds which must be broken, and the mismatched sticky ends make no base-pair bonds.

There are two free parameters in this model, both of which are dimensionless free energies: $G_{mc} > 0$ measures the entropic cost of putting a tile at a binding site and depends on the tile concentration, $G_{se} > 0$ measures the free energy cost of breaking a single strength-1 bond. Under this model, we can approximate the forward and reverse rates for each of the tile-supertile reactions in the process of self-assembly of DNA tiles as follows:

The rate of addition of a tile to a supertile, f, is $pe^{-G_{mc}}$.

The rate of dissociation of a tile from a supertile, r_b, is $pe^{-bG_{se}}$, where b is the strength with which the tile is attached to the supertiles.

The parameter p simply gives us the time scale for the self-assembly.

Winfree suggests using G_{mc} just a little smaller than $2G_{se}$ for self-assembly at temperature two. We use the same operating region.

3 An Illustrative Example

While the ideas that we develop in this section are applicable to general self-assemblies, a simple one dimensional example will be used for illustrative purposes. The tile system is one that can compute the parity of a bit string and we will refer to it as the parity system. The tiles are essentially a simplification of the tiles in the Sierpinski tile system [10] and are obtained by making the top side of each tile in the Sierpinski system inert. The tiles for the parity system are illustrated below in figure 1(a). The temperature is 2. The "input" will consist of a structure of $n+2$ tiles. The "input" tiles are assumed to be arranged in two rows. The bottom row has $n+1$ tiles. The rightmost tile on the bottom row is inert on the right, the leftmost is inert on the left, and they are all inert on the bottom. Each tile in the bottom row except the leftmost has a glue labeled either 0 or 1 on the top. The second row has just one input tile, sitting on top of the leftmost tile in the bottom row. This second row tile is inert on the left and the top, and has a glue labeled 0 on the right. Thus the input codes a string of n bits. With this input, the tiles in the parity system will form a layer covering the n exposed glues in the bottom row. Further, the rightmost tile in the top row will leave a glue labeled 0 on the right if the parity of the bit string is 0 (i.e. the number of ones is even) and 1 otherwise. Figure 1(b) illustrates this construction for $n = 4$ and the input string 1111. The glues are written on the edges of the tiles and the input tiles are shaded.

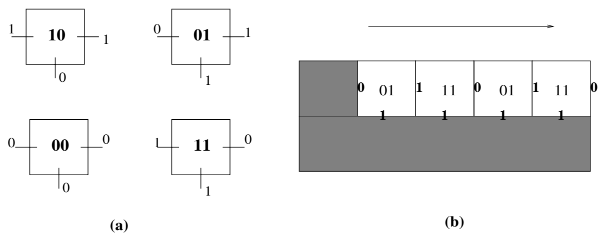

Fig. 1. (a) The parity tile system. (b) Illustrating the action of the parity tile system on the ”input” string 1111. The arrow at the top represents the order in which tiles must attach in the absence of errors

In this setting, tiles in the top row attach from left to right, if there are no errors. Hence, in the absence of errors, there is always a correct “next” position

3.1 Growth Errors and the Winfree-Bekbolatov Proof-Reading System

An error is said to be a “growth” [11] error if an incorrect tile attaches in the next position. The proof-reading approach of Winfree and Bekbolatov [11] can correct such errors by using redundancy. They replace each tile in the system with four tiles, arranged in a 2×2 block. Figure 2(b) depicts the four tiles that replace a 10 tile. The glues internal to the block are all unique. This added redundancy results in resilience to growth errors. The details are described in their paper.

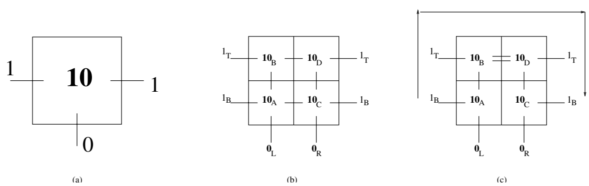

Fig. 2. (a) The original 10 tile. (b) The four proof-reading tiles for the 10 tile, using the construction of Winfree and Bekbolatov [11]. (c) The snaked proof-reading tiles for the parity tile system. The internal glues are all unique to the 2×2 block corresponding to the 10 tile. Notice that there is no glue on the right side of 10_A or the left side of 10_C and that the glue between the top two tiles is of strength 2. This means that the assembly process doubles or “snakes” back onto itself, as demonstrated by the arrow

3.2 Nucleation Errors and Improved Proof-Reading

However, there is another, more insidious kind of error that can happen. A tile may attach at a position other than the correct “next” position using just a strength one glue. This would be the incorrect tile, and hence an error with probability 50%, and such an error will propagate to the right ad infinitum even if we are using the proof-reading tile set of Winfree and Bekbolatov. We call such errors “nucleation” errors[1]. In more

[1] Winfree and Bekbolatov call these facet roughening errors and reserve the term nucleation errors for another phenomenon.

complicated systems, these errors can also happen on the boundary of a completed assembly, making it very hard to precisely control the size of an assembly. Both growth errors and nucleation errors are caused by what we term an insufficient attachment – the attachment of a tile to an existing assembly using a total glue strength of only 1 (even though the temperature is 2) and then being "stabilized" (i.e. held by strength 2) by another tile attaching in the vicinity. Insufficient attachments are unlikely at any given site (say they happen with probability x) but over the course of n attachments, the probability of getting at least one insufficient attachment may become as large as $O(nx)$. We will now show a design that requires two insufficient attachments in close proximity to have an error that can propagate, and significantly reduces the chances of getting an error (either growth or nucleation). Figure 2(c) shows the 2×2 block that replaces a single tile (say tile 10), and the arrow shows the order in which the sub-tiles attach at a site when there have been no insufficient attachments. Notice that there is no glue between tiles 10_A and 10_C. This is what prevents nucleation errors from propagating without another insufficient attachment. We call this the "snaked" proof-reading system, since the assembly process for a block doubles back on itself.

It is easy to show that the above approach can be extended to arbitrary $k \times k$ sized blocks, to get lower and lower error rates. The above idea can also be extended to Sierpinski tile systems [10] and counters [8, 2], though for technical reasons, a 3×3 block is needed at a minimum to take care of nucleation errors in these more complicated systems. Detailed analysis is given in section 4. However, the following lemmas are useful to illustrate the kind of improvements we can expect to get. The quantities f, r and G_{se} are as defined in section 2.2. An *insufficient attachment* at temperature two is the process that a tile attaches with strength one, but, before it falls off, another tile attaches right next to it and both tiles are held by strength at least two.

Lemma 1. *The rate at which an insufficient attachment happens at any location in a growing assembly is* $\frac{f^2}{r} e^{-G_{se}} = O(e^{-3G_{se}})$.

Proof. The rate of an insufficient attachment can be modeled as the Markov Chain shown in figure 3. For a nucleation error to happen, first a single tile must attach (at rate f). The fall-off rate of the first tile is $re^{G_{se}}$ and the rate at which a second tile can come and attach to the first tile is f. After the second tile attaches, an insufficient attachment has happened. So the overall rate of an insufficient attachment is $f * \frac{f}{f+re^{G_{se}}} \approx \frac{f^2}{r} e^{-G_{se}}$

Without proof-reading, or even using the proof-reading system of Winfree and Bekbolatov, a single insufficient attachment can cause a nucleation error, and hence the

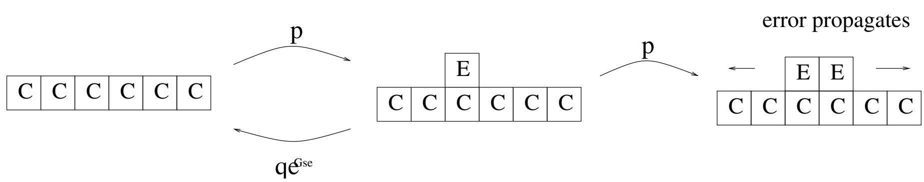

Fig. 3. The C tiles represent the existing assembly, and the E tiles are new erroneous tiles

rate of nucleation error at any location is also $O(e^{-3G_{se}})$. The next lemma shows the improvement obtained using our snaked proof-reading system. The difference is even more pronounced if we compare the nucleation error rate to the growth rate, which is a natural measurement unit in this system. The ratio of the nucleation error rate to the growth rate is $O(e^{-G_{se}})$ in the original proof-reading system, whereas it is $O(e^{-2G_{se}})$ in our system, a quadratic improvement.

Lemma 2. *The rate at which a nucleation error takes place in our snaked proof-reading system is $O(e^{-4G_{se}})$.*

Proof. In the snaked system, two insufficient attachments need to happen next to each other for a nucleation error to occur. According to lemma 1, the first insufficient attachment happens at rate $O(e^{-3G_{se}})$. After the first insufficient attachment, the error will eventually be corrected unless another insufficient attachment happens next to the first. The second insufficient attachment happens at rate $O(e^{-3G_{se}})$; but the earlier insufficient attachment gets "corrected" at rate $O(e^{-2G_{se}})$ (remember that $a \approx 1$ and hence a tile attached with strength 2 falls off at roughly the growth rate). Hence, the probability of another insufficient attachment taking place before the previous insufficient attachment gets reversed is $O(e^{-G_{se}})$, bringing the nucleation error rate down to $O(e^{-4G_{se}})$.

For growth errors, the proof-reading system of Winfree and Bekbolatov achieves a reduced error rate of $O(e^{-4G_{se}})$, a property preserved by our modification.

4 The General Snaked Proofreading System

The system shown in the previous section only works for prevention of nucleation errors in one direction (west to east). The system we describe in this section can improve any rectilinear tile system [2] and prevents nucleation errors in both growth directions.

First, we look at rectilinear systems in which all glues have strength 1. To improve this kind of system, each tile T in the original system is replaced by a $2k \times 2k$ block ($k \geq 2$) $T_{1,1}$, $T_{1,2}$, $\ldots$, $T_{2k,2k}$. Each glue G_i in the original system is replaced by $2k$ glues $G_{i,1}, G_{i,2}, \ldots, G_{i,2k}$ with strength 1 on the corresponding boundary of the block. All glues internal to the block have strength 1 except the following:

1. The east sides of tiles $T_{1,2i-1}$ are inert, as well as the west sides of tiles $T_{1,2i}$ for $i = 1, 2, \ldots, k-1$.
2. The north sides of tiles $T_{2i,1}$ are inert, as well as the south sides of tiles $T_{1,2i+1}$ for $i = 1, 2, \ldots, k-1$.
3. The glues on the north sides of tiles $T_{2i,2i+1}$ have strength 2, as well as the glues on the south sides of tiles $T_{2i+1,2i+1}$ for $i = 1, 2, \ldots, k-1$.
4. The glues on the east sides of tiles $T_{2i,2i-1}$ have strength 2, as well as the glues on the west sides of tiles $T_{2i,2i}$ for $i = 1, 2, \ldots, k$.

[2] A rectilinear tile system is one where growth occurs in a rectilinear fashion - from south to north and from west to east.

5. The east side of the tile $T_{2k-2,2k-1}$ is inert, as well as the west side of the tile $T_{2k-2,2k}$.
6. The glue on the north side of the tile $T_{2k-2,2k}$ has strength 2, as well as the south side of the tile $T_{2k-1,2k}$.

The glues internal to the block are unique to that block and don't appear on any other blocks. Informally, the blocks attach to each other using the same logic as the original system.

An illustrative example with $k = 2$ is shown in figure 4(a). The numbering of the tiles in figure 4(b) denotes the sequence of the tile attachment in the assembly process. It is worth noticing that all the tiles on the northern and eastern side of the block are held by strength at least 3. So whenever all the tiles on a block are attached, it is unlikely for them to fall off.

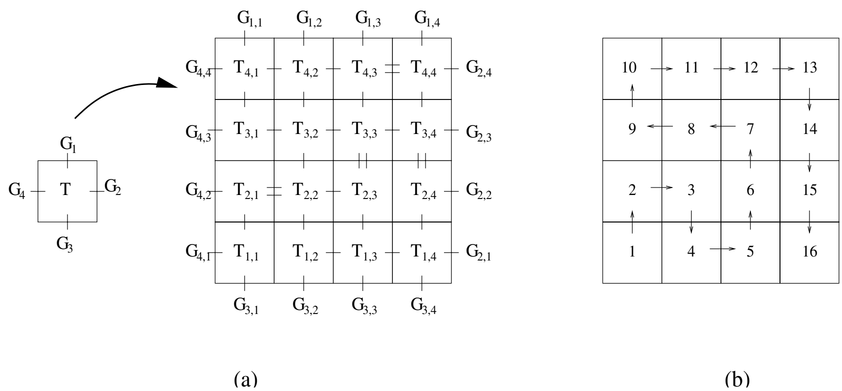

Fig. 4. (a) The structure of 4x4 block. (b) The order of the growth

Recall that f denoted the forward rate of a tile attaching, and r denotes the backward rate of a tile held by strength 2 falling off. In the rest of the section, we assume that $f = r$ and tiles held by strength three do not fall off. We need to make this assumption for our proof to go through, but we don't believe they are necessary.

Here are some definitions we will use in this section: a *k-bottleneck* is a connected structure which requires at least k insufficient attachments to form. A *block error* occurs if all the tiles in a block have attached and are all incorrect (compared to perfect growth).It is easy to prove that a block error is just an example of a k-bottleneck.

We are going to consider an idealized system where the south and west boundary is already assembled and the tiles in the square are going to assemble in a rectilinear fashion. The following theorems represent our main analytical result:

Theorem 1. *With a $2k \times 2k$ snaked tile system (for some fixed k), assuming we can set $e^{G_{se}}$ to be $O(n^{\frac{2}{k}})$, an $n \times n$ square of blocks can be assembled in time $O(n^{1+\frac{4}{k}})$ and with high probability, no block errors happen $\Omega(n^{1+\frac{4}{k}})$ time after that.*

Theorem 2. *With a $2k \times 2k$ snaked tile system, $k = O(\log n)$, assuming that we can set $e^{G_{se}}$ to be $O(k^6)$, an $n \times n$ square of blocks can be assembled in time $\widetilde{O}(n)$ and with high probability, no block errors happen for $\widetilde{\Omega}(n)$ time after that.*

Here, the $\widetilde{O}$ notation hides factors which are polynomial in k and $\log n$. Informally, Theorems 1 and 2 say that snaked proofreading results in tile systems which assembly quickly and remain stable for a long time.

In fact, we believe that our scheme achieves good performance without having to set $e^{G_{se}}$ to be as high as $O(k^6)$ and the ratio between forward and backward rate can be set to some constant for getting a good performance. The simulation results are described at the end of this section confirm our intuition.

4.1 Proof of the Main Result

Since all tiles in a correctly attached block are held by strength three, once a correct block attaches at a location, none of its tiles ever fall off. So, it suffices to only consider the perimeter of the supertile. For ease of exposition, we are going to focus on errors that happen on the east edge of the assembly.

Lemma 3. *Consider any connected structure caused by m insufficient attachments $(1 \le m \le k)$. Then the width of the structure can be at most $2m$, and the height of the structure can be at most $2k$ (i.e., this connected structure can only span two blocks). This structure will fall off in expected time $O(\frac{k^5}{r})$ unless there's a block error somewhere in the assembly or an insufficient attachment happens within the (at most two) blocks spanned by the structure.*

Proof Outline: The proof of this lemma involves a lot of technical details. Due to space constraints, we only present a sketch in this version. In the structure of $2k \times 2k$ snaked tiles, all the glues between the $(2i)$-th row and $(2i+1)$-th row have strength 1 for all i. So, to increase the width from $2i$ to $2i + 1$, we must have at least one insufficient attachment. So, with m insufficient attachments, the width of the structure can be at most $2m$. Using similar arguments, the height of the structure can be at most $2k$. Also, the attached tiles can be partitioned into $O(k)$ parts. Each of these parts can be viewed as a $2 \times O(k)$ rectangle with every internal glue having strength 1. The process of tiles attaching to and detaching from each rectangle can be modeled using two orthogonal random walks and hence, each rectangle will fall off in expected time $O(\frac{k^4}{r})$. The different rectangles can fall off sequentially, and after one rectangle falls off completely, none of its tiles will attach again unless an insufficient attachment happens. Thus, the structure will fall off in expected time $O(\frac{k^5}{r})$ unless there's a block error (anywhere in the assembly) or an insufficient attachment happens (within the two blocks) before the structure has a chance to fall off. □

Theorem 3. *Assume that we use a $2k \times 2k$ snaked tile system and $G_{mc} = 2G_{se}$. Then for any ϵ, there exists a constant c such that, with probability $1 - \epsilon$, no k-bottleneck will happen at a specific location within time $c\frac{1}{f}e^{G_{se}}(\frac{e^{-G_{se}}+1/k^6}{e^{-G_{se}}})^{k-1}$.*

Proof. By definition, k insufficient attachments are required before a k-bottleneck happens. After i insufficient attachments take place, one of the following is going to happen:

- One more insufficient attachment. Consider any structure X caused by i insufficient attachments. By Lemma 3, the size of X cannot exceed two blocks, hence the number of insufficient attachment locations that can cause this structure to grow larger is at most $4k$. So, the rate of the $(i+1)$-th insufficient attachment happening is at most $4kfe^{-G_{se}}$.
- All the attached tiles fall off. By Lemma 3, the expected time for all the attached tiles to fall off is $O(\frac{k^5}{r})$

So, after i insufficient attachments happen, the probability of the $(i+1)$-th insufficient attachment happening before all tiles fall off is $O(\frac{kfe^{-G_{se}}}{kfe^{-G_{se}}+r/k^5}) = O(\frac{e^{-G_{se}}}{e^{-G_{se}}+1/k^6})$. So, after the first insufficient attachment takes place, the probability of a k-bottleneck happening before all the attached tiles fall off is less than $O((\frac{e^{-G_{se}}}{e^{-G_{se}}+1/k^6})^{k-1})$. As shown in Lemma 1, the expected time for the first insufficient attachment is $O(\frac{1}{f}e^{G_{se}})$. So, the expected time for a k-bottleneck to happen at a certain location is at most $O(\frac{1}{f}e^{G_{se}}(\frac{e^{-G_{se}}+1/k^6}{e^{-G_{se}}})^{k-1})$. Hence, for any small ϵ, we can find a constant c such that, with probability $1-\epsilon$, no k-bottleneck will happen at a specific location within time $c\frac{1}{f}e^{G_{se}}(\frac{e^{-G_{se}}+1/k^6}{e^{-G_{se}}})^{k-1}$.

Theorem 4. *If we assume there are no k-bottlenecks, and the rate of insufficient attachments is at most $O(\frac{f}{k^6})$, then an $n \times n$ square of $2k \times 2k$ snaked tile blocks can be assembled in expected time $O(\frac{k^5 n}{f})$.*

Proof. With the snaked tile system, after all the tiles in a block attach, all the tiles are held by strength at least 3 and will never fall off. Using the running time analysis technique of Adleman *et al.* [2], the system finishes in expected time $O(n \times T_B)$, where n is the size of the terminal shape and T_B is the expected time for a block to assemble. Without presence of k-bottlenecks, when we want to assemble a block, the erroneous tiles that currently occupy that block are formed by at most $k-1$ insufficient attachments. By Lemma 3, without any further insufficient attachments happening, the erroneous tiles will fall off in time $O(\frac{k^5}{f})$ and the correct block can attach within time $O(\frac{k^4}{f})$. By assumption, the rate of insufficient attachment happening is at most $O(\frac{f}{k^6})$, and there are at most $O(k)$ locations for insufficient attachments to happen and affect this process. So, there's a constant probability that no insufficient attachments will happen during the whole process and thus the time required to assemble a block, T_B, is at most $O(\frac{k^5}{f})$.

Theorems 1 and 2 follow from the above two theorems. Notice that there is a lot of slack in our analysis.

4.2 Simulation Results

We use the simulation program xgrow written by Winfree *et al.* [14].

We first use three different systems to build a square of 20×20 blocks with the Sierpinski pattern [10]. The column "snaked" refers to the system described in this paper; the column "proofreading" corresponds to the original proofreading system described by Winfree and Bekbolatov; the column "original" refers to the system without any error correction. The results are summarized in table 1. The block size for our snaked system as well as the original proofreading system is 4×4. Similar results were observed for a wide range of simulation scenarios. As is clear, our snaked tile system has a much lower error rate, has a much higher stability time, and is only two-three times slower than the proofreading system of Winfree and Bekbolatov [11]. The original system only needs to assemble a much smaller structure (since it uses a 1×1 block); hence the "reversal" in the error-rates for the original and the proofreading system in the second simulation.

Table 1. Assembling a 20×20 Sierpinski block. A stability time of 0 indicates that the final square became unstable (i.e., an extra block of tiles attached on the periphery of the desired supertile) even before the complete supertile formed. The values represent averages over 100 runs

	$G_{mc} = 15, G_{se} = 7.8$			$G_{mc} = 15, G_{se} = 8.0$		
	Original	Proofreading	Snaked	Original	Proofreading	Snaked
Time to assemble (seconds)	550	2230	6020	350	1750	3780
Error Probability	52%	24%	0%	63%	75%	0%
Time it remains stable (seconds) after completion	0	0	>400000	0	0	5700

Also, for the 2×2 snaked tile system and the original proofreading system, we took a straight line boundary of 200 tiles (i.e., 100 blocks) and tested the average time (in seconds; virtual time) for a block error to happen under different G_{se}'s. Here, $2G_{se} - G_{mc}$ is set to be 0.2. Theoretically, the expected time for a block error to happen in the snaked tile system is $O(e^{4G_{se}})$, and the expected time for a block error to happen in the proofreading system is $O(e^{3G_{se}})$. The result is shown in figure 5(a); the y-axis uses a log-scale. Clearly, the slope of the curve for the snaked tile system confirms our analysis – the slope is very close to 4, and significantly more than the slope for the original proofreading system. For the larger values of G_{se}, we could only plot the results for the original proof-reading system, since the simulator did not report any errors with the snaked tile system for the time scales over which we conducted the simulation.

We also tested the error rate for parity systems of different seed lengths. We called an experiment an error if the final supertile was different from the one we expect in the absence of errors. We used $G_{se} = 7.0, G_{mc} = 13.6$. The result is shown in figure 5(b); again, a significant reduction in error rate is observed. For both figures 5(a) and 5(b), qualitatively similar results were observed for widely varying simulation parameters.

Our simulation results show that our analysis is very close to reality even without idealized parameter conditions. For example, we did not use $G_{mc} = 2G_{se}$ but instead used G_{mc} slightly smaller than $2G_{se}$ as suggested by Winfree [10]. Also, the simulator

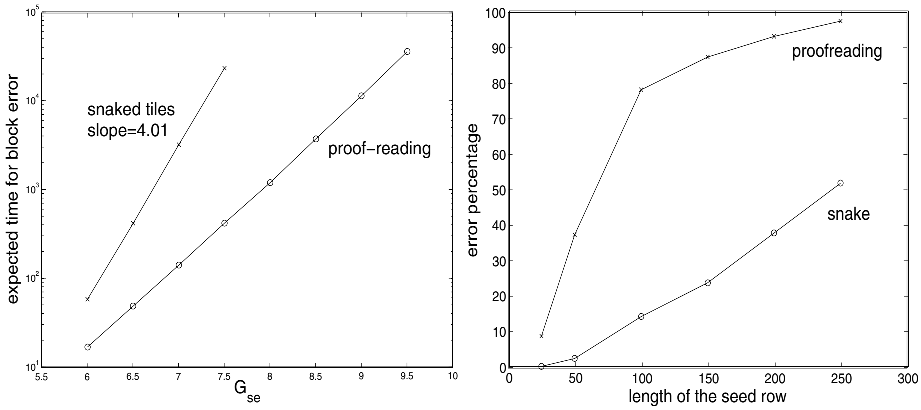

Fig. 5. (a) The first figure shows the relation between the average time for a block error to happen and G_{se} (average over 100 runs). (b) The second figure shows that the error rate for snaked tiles is much smaller than proofreading tiles (average over 200 runs)

allows tiles held by strength 3 to fall off, contrary to our assumption. Thus, we believe that our snaked system works much better (and under a much wider set of conditions) than we have been able to formally prove.

5 Future Directions

It would be interesting to extend our analysis to remove some of our assumptions. Also, we believe that the total assembly time for our system should just be $O(k^2n)$ for assembling an $n \times n$ square using $k \times k$ snaked blocks. One of the biggest bottlenecks in proving this bound is an analysis of the assembly time of an $n \times n$ square assuming that there are no errors but that the system is reversible, i.e., tiles can both attach and detach. We believe that the assembly time for this system should be $O(n)$ along the lines of the irreversible system [2], but have not yet been able to prove it.

Acknowledgments

We would like to thank Qi Cheng and Erik Winfree for many useful discussions. We would also like to thank Erik Winfree for loaning us his tile simulator, xgrow, which was an indispensable aid in this research.

References

1. L. Adleman. Towards a mathematical theory of self-assembly. Technical Report 00-722, Department of Computer Science, University of Southern California, 2000.
2. L. Adleman, Q. Cheng, A. Goel, and M.-D. Huang. Running time and program size for self-assembled squares. In *Proceedings of the thirty-third annual ACM symposium on Theory of computing*, pages 740–748. ACM Press, 2001.

3. L. Adleman, Q. Cheng, A. Goel, M.-D. Huang, D. Kempe, P. Moisset de Espans, and P. Rothemund. Combinatorial optimization problems in self-assembly. In *Proceedings of the thiry-fourth annual ACM symposium on Theory of computing*, pages 23–32. ACM Press, 2002.
4. H. Chen, Q. Cheng, A. Goel, M.-D. Huang, and P. Moisset de Espans. Invadable self-assembly: Combining robustness with efficiency. In *Proceedings of the Fifteenth Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 883–892, 2004.
5. M. Lagoudakis and T. LaBean. 2D DNA self-assembly for satisfiability. In *Proceedings of the 5th DIMACS Workshop on DNA Based Computers in DIMACS Series in Discrete Mathematics and Theoretical Computer Science*, volume 54. MIT: Cambridge, 1999.
6. J. Reif. Local parallel biomolecular computation. In H. Rubin, editor, *Third Annual DIMACS Workshop on DNA Based Computers, DIMACS Series in Discrete Mathematics and Theoretical Computer Science*, 1998.
7. P. Rothemund. *Theory and Experiments in Algorithmic Self-Assembly*. PhD thesis, University of Southern California, 2001.
8. P. Rothemund and E. Winfree. The program-size complexity of self-assembled squares (extended abstract). In *Proceedings of the thirty-second annual ACM symposium on Theory of computing*, pages 459–468. ACM Press, 2000.
9. H. Wang. Proving theorems by pattern recognition II. Bell Systems Technical Journal, 1961. 40:1-42.
10. E. Winfree. *Algorithmic Self-Assembly of DNA*. PhD thesis, California Institute of Technology, Pasadena, 1998.
11. E. Winfree and R. Bekbolatov. Proofreading tile sets: Error correction for algorithmic self-assembly. In *Proceedings of the Ninth International Meeting on DNA Based Computers*. Madison, Wisconsin, June 2003.
12. E. Winfree, F. Liu, L. Wenzler, and N. Seeman. Design and self-assembly of two-dimensional DNA crystals. *Nature*, (394):539–544, Aug 1998.
13. E. Winfree, X. Yang, and N. Seeman. Universal computation via self-assembly of DNA: Some theory and experiments. In *Proceedings of the Second Annual Meeting on DNA Based Computers*. Princeton University, June 1996.
14. E. Winfree *et al.*. The xgrow simulator. *http://www.dna.caltech.edu/Xgrow/xgrow_www.html*.
15. B. Yurke, A. Turberfield, A. Mills Jr, F. Simmel, and J. Neumann. A DNA-fuelled molecular machine made of DNA. *Nature*, (406):605–608, Aug 2000.