Running Time and Program Size for Self-assembled Squares*

Leonard Adleman†

‡
University of Oklahoma

Ashish Goel §
Stanford University

Ming-Deh Huang ¶
University of Southern California

Pablo Moisset de Espanés ‖
University of Southern California

University of Southern California

Abstract

Recently Rothemund and Winfree [8] have considered the program size complexity of constructing squares by self-assembly. Here, we consider the time complexity of such constructions using a natural generalization of the Tile Assembly Model defined in [8]. In the generalized model, the Rothemund-Winfree construction of $n \times n$ squares requires time $\Theta(n \log n)$ and program size $\Theta(\log n)$. We present a new construction for assembling $n \times n$ squares which uses optimal time $\Theta(n)$ and program size $\Theta(\frac{\log n}{\log \log n})$. This program size is also optimal since it matches the bound dictated by Kolmogorov complexity. Our improved time is achieved by demonstrating a set of tiles for parallel self-assembly of binary counters. Our improved program size is achieved by demonstrating that self-assembling systems can compute changes in the base representation of numbers. Self-assembly is emerging as a useful paradigm for computation. In addition the development of a computational theory of self-assembly promises to provide a new conduit by which results and methods of theoretical computer science might be applied to problems of interest in biology and the physical sciences.

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 has been suggested that self-assembly will

*A preliminary version of this paper appeared in the 33rd ACM Symposium on Theory of Computing, 2001.

†Professor of Computer Science and Molecular Biology, University of Southern California. Research supported by grants from NASA/JPL, NSF, ONR, and DARPA. Email: adleman@usc.edu

‡Qi Cheng is at the Department of Computer Science, University of Oklahoma. Research supported by an NSF CAREER award. Email: qcheng@ou.edu.

§Ashish Goel is at the Department of Management Science and Engineering and (by courtesy) Computer Science, Stanford University, Terman 311, Stanford CA 94305. Email: ashishg@stanford.edu. Research supported by NSF CAREER Award 0133968 and a grant from NASA.

¶Ming-Deh Huang is at the Department of Computer Science, University of Southern California. Email: huang@usc.edu. Research supported in part by a grant from NASA and by NSF Grant CR-9820778.

‖Pablo Moisset de Espanés is at the Department of Computer Science, University of Southern California. Email: pmoisset@usc.edu. Research supported in part by a grant from NASA/JPL.

ultimately become an important technology, enabling the fabrication of great quantities of intricate objects such as computer circuits from inexpensive components such as DNA and inorganic nanocrystals. Despite its importance, self-assembly is poorly understood. Recently, work related to DNA computation has led to experimental systems for the investigation of self-assembly and its relation to computation [10, 6, 4, 3]. In addition certain theoretical aspects of self-assembly have been considered. Winfree [10, 11] proved that self-assembling tile systems in a plane are capable of doing universal computation, and when restricted to a line are exactly as powerful as discrete finite automata. Adleman [1] proposed a mathematical model of self-assembly and analyzed the time complexity of linear polymerization. Rothemund and Winfree [8] proposed the Tile Assembly Model of self-assembly and studied the program size complexity (the number of different tile-types used) of constructing $n \times n$ squares. In this paper we extend the Tile Assembly Model to include the time complexity of the assembly process. We then demonstrate a system of tiles which assembles into an $n \times n$ square while simultaneously achieving optimal ($\Theta(n)$) time and optimal program size ($\Theta(\log n/ \log\log n)$). In contrast, the system proposed by Rothemund and Winfree takes time $\Theta(n \log n)$ and uses $\Theta(\log n)$ program-size in the worst case. It is our hope that an understanding of simple self-assembling systems will pave the way for a general theory of self-assembly.

To achieve the improved time complexity, we show how a binary counter that counts from 0 to n can be assembled in expected time $\Theta(n)$, as opposed to the $\Theta(n \log n)$ time for the same assembly in the work of Rothemund and Winfree. Further, the assembly time has an exponentially decaying tail. The number of increment steps required to count from 0 to n is exactly n. Thus our construction takes an amortized time of $\Theta(1)$ for an increment step, even though each increment step requires an addition of $\Theta(\log n)$ new tiles. This is made possible by the parallelism inherent in our tile-assembly system. We also observe that no system can result in a square being assembled in expected time less than $\Omega(n)$ in the model suggested by Rothemund and Winfree and expanded by us in this paper.

In order to count to n, a $\log n$-bit counter is required. Most of the $\Omega \log n)$ tiles in the construction of Rothemund and Winfree were used to produce the first row (the seed row) in the counter. The increment steps in their counter assembly (and in ours) can be performed using a constant number of tiles, and the fully assembled counter, which is roughly a $\log n \times n$ rectangle can be completed into a square using a constant number of tiles. To eliminate the need to produce a seed-row that is $\log n$ bits long, we represent the number n in base $\Theta(\log n/ \log\log n)$ using only $\Theta(\log n/ \log\log n)$ digits. Thus if the counter were to use base $\Theta(\log n/ \log\log n)$ rather than binary, only $\Theta(\log n/ \log\log n)$ tiles would be required to assemble a counter. This observation immediately allows us to reduce the number of tiles (the program size) required to assemble a square to $\Theta(\log n/ \log\log n)$, which matches the lower bound dictated by Kolmogorov complexity. Unfortunately, this program size reduction comes at the cost of increased assembly time. To remedy this, we introduce the notion of base conversion. We start out with a row of $\Theta(\log n/ \log\log n)$ tiles representing a number between 0 and n in base $\Theta(\log n/ \log\log n)$. We then convert this number into binary using a self-assembly process that simulates base conversion. Now the counting process can take place in binary, allowing us to achieve the optimal time complexity and optimal program-size complexity *simultaneously*.

Section 2 presents a natural extension of the tile-assembly model of Rothemund and Winfree to include the time-complexity of self-assembly. Section 3 shows how a $\log n \times n$ counter can be assembled in time $\Theta(n)$ and section 5 explains how base conversion can be simulated in the tile-assembly model. Section 6 sketches how an entire $n \times n$ square can be constructed using the tools outlined in sections 3 and 5. Section 7 describes a more general running time analysis technique. Finally, we present some open problems and conclusions in section 8.

2 Adding time complexity to the Tile Assembly Model

The Tile Assembly Model was originally proposed by Rothemund and Winfree [8]. It extends the theoretical model of tiling by Wang [9] to include a mechanism for growth based on the physics of molecular self-assembly. Since self-assembly is an emerging research field, there no widely accepted terminology. For this reason the notation we use in this paper is somewhat ad-hoc and may not be appropriate to describe other self-assembly problems such as the combinatorial optimizations presented in [2].

Informally each unit of an assembly is a square with glues of various types on each edge. The tile ”floats” on a two dimensional plane and when two tiles collide they stick if their abutting sides have compatible glues.

Formally, a tile is an oriented unit square with the north, east, south and west edges labeled from some alphabet Σ of glues. We begin with a triple $\langle T, g, \tau \rangle$ 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 $\mathbf{N}$, where Σ is the set of edge labels and $\mathbf{N}$ is the set of natural numbers. It is assumed that $null \in \Sigma$, $g(x, y) = g(y, x)$ for $x, y \in \Sigma$, and $g(null, x) = 0$ for all $x \in \Sigma$. For each tile $i \in T$, the labels of its four edges are denoted $\sigma_N(i)$, $\sigma_E(i)$, $\sigma_S(i)$, and $\sigma_W(i)$.

Given $p = (x, y), p' = (x', y') \in \mathbf{Z}^2$, we say p and p' are *position adjacent* iff $|x - x'| + |y - y'| = 1$. A *shape* S is a finite subset of $\mathbf{Z}^2$, such that for all pairs of positions p, p' in S, either $p = p'$ or there exists a sequence of positions $p_1, p_2, \cdots, p_n$ such that $p_1 = p$, $p_n = p'$ and for all $1 \leq k < n$, p_k and p_{k+1} are position adjacent. A *configuration* is a map from $\mathbf{Z}^2$ to $T \cup \{\mathbf{empty}\}$. For $t \in T$, $\Gamma_t^{(x,y)}$ is the configuration such that $\Gamma_t^{(x,y)}(i, j) = t$ iff $(i, j) = (x, y)$ and **empty** otherwise. Let C and D be two configurations. Suppose there exist some $i \in T$ and $(x, y) \in \mathbf{Z}^2$ such that $C(x, y) = \mathbf{empty}$, $D = C$ except at (x, y), $D(x, y) = i$, and

$$g(\sigma_E(i), \sigma_W(D(x+1, y)) + g(\sigma_W(i), \sigma_E(D(x-1, y)) +$$
$$g(\sigma_N(i), \sigma_S(D(x, y+1)) + g(\sigma_S(i), \sigma_N(D(x, y-1)) \geq \tau.$$

Then we say that the position (x, y) in C is *attachable*, and we write $C \rightarrow_{\mathbf{T}} D$ to denote the transition from C to D in attaching tile i to C at position (x, y). Informally, $C \rightarrow_{\mathbf{T}} D$ iff D can be obtained from C by adding a tile to it such that the total strength of interaction in adding the tile to C is at least τ.

A *seed configuration* is a configuration Γ of T such that $\{p | \Gamma(p) \neq \mathbf{empty}\}$ is a shape.

A *tile system* is a quadruple $\mathbf{T} = \langle T, S, g, \tau \rangle$, where T, g, τ are as above and S is a set of seed configurations called *seed s-tiles*.

We define the notion of a *supertile* (s-tile in brief) recursively as follows:

1. Every element in S is an s-tile.
2. For $t \in T$, $\Gamma_t^{(x,y)}$ is an s-tile.
3. if $C \rightarrow_{\mathbf{T}} D$ and C is an s-tile, then D is also an s-tile.

From this definition, it is clear that for every s-tile A, $\{p | A(p) \neq \mathbf{empty}\}$ is a shape, and we will call that set the *shape of A* and we will use the notation $[A]$ to refer to it. Write $A \rightarrow_{\mathbf{T}} B$ for s-tiles A and B iff there exist $a \in A$ and $b \in B$ such that $a \rightarrow_{\mathbf{T}} b$.

Let $\rightarrow_{\mathbf{T}}^*$ denote the reflexive transitive closure of $\rightarrow_{\mathbf{T}}$. A *derived supertile* of the tile system $\mathbf{T}$ is an s-tile such that $s \rightarrow_{\mathbf{T}}^* A$ for some $s \in S$. A *terminal supertile* of the tile system $\mathbf{T}$ is a derived supertile A

such that there is no s-tile B, different from A, such that $A \rightarrow^*_{\mathbf{T}} B$. 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. We also say that for a given shape S, a tile system *uniquely produces* S iff it uniquely produces some supertile A and $[A] = S$.

Given a tile system **T** which uniquely produces A, we say that the program size complexity of the system is $|T|$ i.e. the number of tile types.

In this paper, we adopt the restriction, suggested by Rothemund and Winfree [8], that S contains a single seed s, and that $g(\alpha, \beta) = 0$ for $\alpha, \beta \in \Sigma$ with $\alpha \neq \beta$. For a discussion of the lower bound on program-size in the absence of the latter restriction, see open problem #2 in section 8. If the seed s consists of a single tile we will say the system is a *unit-seed* tile system.

We now introduce the definition of the time complexity of self-assembly. A similar definition has also been suggested by Winfree [12]. We associate with each tile $i \in T$ a positive probability $P(i)$, such that $\sum_{i \in T} P(i) = 1$. P is called a *concentrations function* for T. We assume that the tile system has an infinite supply of each tile, and $P(i)$ models the concentration of tile i in the system – the probability that tile i is chosen when a tile is drawn at random. Now self-assembly of the tile system $\mathbf{T}$ corresponds to a continuous time Markov process where the states are in a one-one correspondence with derived s-tiles, and the initial state corresponds to the seed s. There is a transition of state B to C iff $B \rightarrow_{\mathbf{T}} C$, and the rate of the transition is $P(i)$ if C is obtained from B by adding a tile i. Suppose the tile system uniquely produces an s-tile A_T. It would follow that A_T is the unique sink state. Given the Markov process, the time for reaching A_T from s is a random variable. The time complexity for producing A_T from s is defined as the expected value of this random variable.

Informally our definition of time models a system wherein a seed ”floats” in solution encountering tiles at random. The higher the concentration of a particular tile the higher the rate at which it is encountered. When a tile is encountered which has sufficiently strong interaction with the seed, the tile is incorporated. By this process of accretion the seed grows larger and larger.

We say a tile system produces an $n \times n$ square iff it uniquely produces a terminal s-tile which is an $n \times n$ square of tiles. Then the time complexity of producing an $n \times n$ square is the minimum of the time complexity of all the tile systems which produce $n \times n$ squares. The following theorem is immediate from our formal model.

Theorem 2.1 *The time complexity of producing an* $n \times n$ *square is* $\Omega(n)$.

3 Counting up to n in time $\Theta(n)$

The square construction of Rothemund and Winfree [8] occurs in two stages. They first show how to assemble a $\log n \times n$ rectangle, and then extend it into an $n \times n$ square. To assemble the $\log n \times n$ rectangle, they simulate a counter that counts from 1 to n in binary. We are going to use the same general framework, but will replace their counter assembly by a more efficient process. In their counter assembly, only one tile is attachable to the assembly at any given time. In our assembly process, several tiles may be attachable at the same time. This “parallelism” allows us to assemble a counter in time $\Theta(n)$ as opposed to the $\Theta(n \log n)$ assembly time for the counter described by Rothemund and Winfree. We call the process described below the SA-counter.

3.1 The Tile System

We initially assume that $n = 2^K$ for some positive integer K – we will later show how this assumption can be removed. While each tile is completely specified by its four glues, it is convenient for the purpose of exposition to allow tiles to have labels. Each tile has one main label (either 0 or 1) which corresponds to the bit represented by the tile. Further, the east-most zero in a binary number is any is labeled as 0^*. The fully assembled counter is going to be a rectangle, with K tiles in each row. Each row represents a number between 0 and n – this number can be obtained by reading the main labels on the tiles in the row, with the most significant bit being the westmost in the row. We describe the self-assembly process which results in the counter being incremented.

The increment operation uses 8 different tiles as well as K special tiles that appear in the seed row only (see Figure 1). The east-most zero in the i^{th} row (auxiliary label 0^*) is replaced by a 1-tile in the $(i+1)^{st}$ row. Further, the assembly of the $(i+1)^{th}$ row always starts as the attachment of a 1-tile on top of a 0^*-tile. All digits to the east of the east-most zero are replaced by 0-tiles. All digits to the west of the east-most zero are replicated in the new row. If the east-most zero in the i^{th} row is the least significant digit then the position of the west-most zero in the $(i+1)^{st}$ row must be determined by a sequential search running from east to west.

We will assume that the temperature for this counter construction is 2. We now describe the tiles in each of the rows and their glues. All tiles occur with the same probability, p_{SA} which is a constant independent of n.

0-tiles: There are the following 0-tiles: 0^*_E, 0^*_M, 0_Z, 0_C.

1-tiles: 1_E, 1_M, 1_S, 1_C.

The glues and their strengths are depicted pictorially in figure 1.

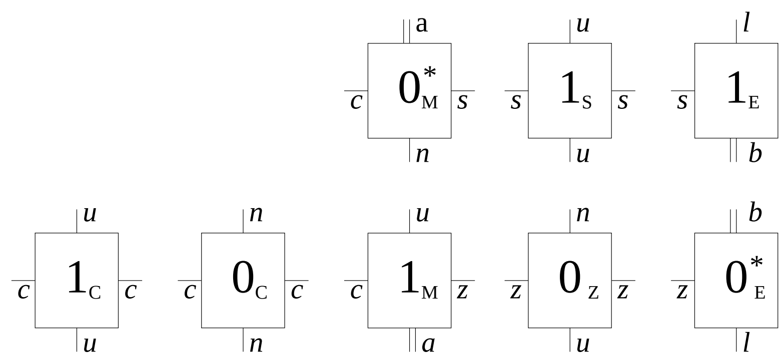

Figure 1: The tiles for the SA-counter. The number of lines jutting from each edge of the tile represent the strength of the bond (1 or 2) and the label on the edges represents the glue. Some edges do not have any glues.

The label suffixes indicate the position in the row each row a tile may occupy and or the function in the assembly.

E: The tile can be only in the east-most column of the counter.

M: The tile cannot be in the east-most column of the counter.

Z: Attach a 0-tile to the east.

C: Copy to the west the digit in the lower row.

S: Search to the west for the position of the first 0-tile in the lower row.

The Seed Row: The seed supertile S_K is a row of K special tiles.

The tile system $\mathbf{T}_{SA}(K)$ has a tile set consisting of all the tiles in figure 1. The glue strength function is as indicated in figure 1. The temperature is 2, and there is a single seed s-tile S_K. We will refer to the Markov chain defined by the tile system as the SA-counter. We will assume that all tiles which are not in S_K have the same constant probability.

We reproduce a definition from [7] with minor modifications in the terminology to make it consistent with the definitions in this paper.

Deterministic row-column (RC) property: Let $\mathbf{T} = \langle T, S, g, \tau \rangle$ be a unit-seed tile system and let Γ be a terminal s-tile of $\mathbf{T}$ and let S be the shape of Γ. Let Let $w_S = \min\{x | \exists y \text{ s.t. } (x, y) \in S\}$, i.e. the x coordinate of the westmost tile. Define e_S, n_S and s_S similarly.

We say Γ has the *C-property* iff the shape of Γ has no holes and for all $w_S \leq x < e_S$ there exists exactly one y such that the strength of the glue between $\Gamma(x, y)$ and $\Gamma(x+1, y)$ is greater or equal than τ. Similarly, we say Γ has the *R-property* iff the shape of Γ has no holes and for all $s_S \leq y < n_S$ there exists exactly one x such that the strength of the glue between $\Gamma(x, y)$ and $\Gamma(x, y+1)$ is greater or equal than τ. We say Γ has the *RC-property* iff it has the R-property and the C-property. We say an attachment of a tile t to a s-tile X at a position p is *deterministic* iff t is the only attachable tile to X at p. We say Γ has the *deterministic RC-property* iff it has the RC-property and there exists a derivation of Γ such that all attachments in that derivation are deterministic.

This theorem, also from [7] will help us to prove correctness of the counter.

Theorem 3.1 *If a unit-seed tile system $\mathbf{T}$ produces an s-tile Γ that has the Strong RC-property, then $\mathbf{T}$ uniquely produces Γ.*

Theorem 3.2 *The tile system $\mathbf{T}_{SA}(K)$ uniquely produces an s-tile which is a $K \times 2^K$ rectangle.*

Proof: Consider a unit-seed tile system $\mathbf{T}'_{SA}(K)$ that results from modifying the K special tiles so $\mathbf{T}'$ can assemble the seed-line starting from the unit seed at the eastmost position. This can be easily accomplished by making all east-west bonds in the seed line of strength 2.

Clearly, there exists a derivation that assembles the rectangle row by row, i.e. no tile is attached in row before the row below is totally assembled. Further, observe that in that derivation all attachments are deterministic. Therefore, Γ has the Strong RC property, and by Theorem 3.1, $\mathbf{T}'_{SA}(K)$ uniquely produces a $K \times n$ rectangle.

By contradiction, consider now the original tile system $\mathbf{T}_{SA}(K)$ and assume $\mathbf{T}_{SA}(K)$ does not produce a $K \times n$ rectangle. That would imply $\mathbf{T}'_{SA}(K)$ does not uniquely produce Γ. ■

Minor modification of the seed row results in a tile system that uniquely produces an s-tile which is a $K \times n$ rectangle for any $n \leq 2^K$.

3.2 Analysis

Our counter construction is more involved than that proposed by Rothemund and Winfree [8] but exploits "parallelism" to speed up the assembly process. The construction of Rothemund and Winfree takes time $O(n \log n)$ in the model of running time described in Section 2.We show that the SA-counter gets assembled in linear time in Section 4). Intuitively, the stronger $\Theta(n)$ bound on the assembly time is due to the fact that a row may start getting assembled even before the previous row is completely assembled.

In spite of having many possible derivations of the rectangle from the seed row, this particular tile system has a property that makes the analysis of the running time easier. We reproduce the definition of *partial order systems* introduced in [2], with minor adaptations.

Assume that we are given a tile system $\mathbf{T}$ and a shape S that is uniquely produced by $\mathbf{T}$. Let A denote the s-tile of $\mathbf{T}$ that has shape S. Let $A(i,j)$ represent the tile-type at position (i,j). Consider a derivation of A and let $t_{i,j}$ represent the time when the tile at position (i,j) attaches to the growing assembly. Define a partial order $\prec$ on the tile positions in S such that $(i,j) \prec (p,q)$ iff $t_{i,j} \leq t_{p,q}$ for all possible derivations of A using $\mathbf{T}$.

Definition: A tile system $\mathbf{T}$ is said to be a *partial order system* iff it uniquely produces a shape S, and if for all adjacent positions (i,j),(x,y) in S, either $(i,j) \prec (p,q)$, or $(x,y) \prec (i,j)$, or the strength of the glues connecting tiles at positions (i,j) and (x,y) is zero.

Note that we can represent the partial order relation as a $DAG\ G = (S,E)$, where $(p,q) \in E$ iff $p \prec q$.

Figure 2 depicts the assembly of a SA-counter counting from 0 to 7 in binary. The thin arrows indicate the order relation in the process. For a arrow, the position at the tail must be filled before the position at the head. Intuitively, a long path in the graph suggests long running time because it implies those positions will be filled sequentially.

Lemma 3.3 *The tile system $\mathbf{T}_{SA}(K)$ is a partial order system.*

Proof: For all rows (other than the seed row), the first tile to be added to that row is the one on top of the 0^* tile in the immediately lower row. This is because only tiles labeled as 0^* have a strength 2 glue on top. Since there is exactly one 0^* tile per row, we observe that no tile can attach to the assembly if the tile immediately to the south is not already in place. This gives the north-south order relation. Inside each row, we note that for all positions p to the west of the 0^* tile, an attachment is impossible if the position immediately to the east of p is empty. Similarly, for all positions p to the east of the 0^* tile, an attachment is impossible if the position immediately to the east of p is empty. This completes the definition of the partial order relation for all pairs of adjacent positions. ■

Let G_n be the DAG representing the partial order relation in an SA-counter counting from 0 to n.

Lemma 3.4 *The length of the longest directed path in G_n is $\Theta(n)$.*

Proof: Let L be length of the longest path in G_n. The longest path is the one containing the most west-east edges. There is exactly one path containing all of them. It is the path containing all the vertices corresponding to the 0^* labeled tiles. Any path going from the bottom row to the top row will traverse exactly $n-1$ north-south edges, hence $L = \Omega(n)$. The number of west-east and east-west edges traversed is easily bounded by adding the distance from the east-most zero to the east-most column for all rows.

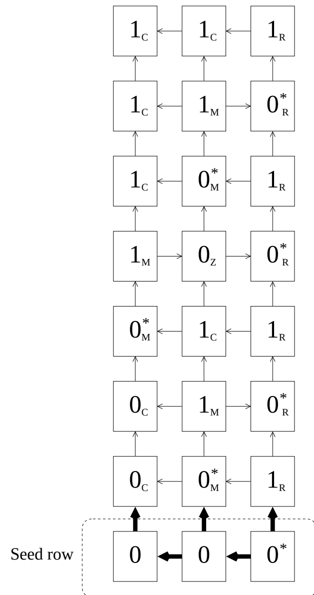

Figure 2: Graph representing the assembly of a counter. The dashed line encloses the seed row. The thin arrows indicate the order relation. The thick arrows represent the edges added to construct an Equivalent Acyclic graph.

$L \leq n - 1 + \sum_{j=1}^{k} (k-j) \times 2^j$

and

$n - 1 + \sum_{j=1}^{k} (k-j) \times 2^j \leq n + k \sum_{i=0}^{k-1} i 2^{k-i} \leq n(1 + \sum_{i=0}^{\infty} i 2^{-i}) = O(n)$

■

Theorem 3.5 *The time complexity for building an SA-counter that counts from* 0 *to* n *is* $\Theta(n)$.

We will present the proof of Theorem 3.5 at the end of Section 4 after we develop some more machinery.

4 Running time Analysis.

For the purpose of our analysis, we transform the self-assembly process into another process which we call the "sentinel" process. The sentinel process does not adhere to the model described in Section 2; in fact there does not seem to be an easy implementation of the sentinel process. However, the sentinel process is more amenable to analysis, and the time for this process to complete is an upper bound (in the stochastic domination sense) on the completion time for the self-assembly process.

Given a directed graph $G = (V, E)$ and vertices v, v' in V, we say v is a *predecessor* of v' iff $(v, v') \in E$. Given a vertex v, we define the *predecessors of v in G*, and we denote it as $Pred_G(v)$, to be the set of all vertices v' such that v' is a predecessor of v.

Equivalent Acyclic Graph (EAG): Let $\mathbf{T} = \langle T, S, g, \tau \rangle$ be a tile system that uniquely produces some s-tile Γ, and let $G = (V, E)$ be a DAG. G is an *Equivalent Acyclic Graph (EAG) for* **T** iff:

1. $V = [\Gamma]$.
2. G has exactly one source.
3. For all edges $(p, p') \in E$, p is position-adjacent to p'.
4. Let σ be the seed s-tile. For all $p \in [\Gamma] - [\sigma]$, for all derived s-tiles Γ' of **T**, if $Pred_G(p) \subset [\Gamma]$ and $p \notin [\Gamma']$, then $\Gamma(p)$ is attachable to Γ' at p.

Note that for a given **T** there may exist more than one EAG.

Lemma 4.1 *For all tile systems* **T** *that uniquely produce an s-tile, there exists an* EAG.

Proof: Let Γ be the terminal s-tile produced by **T**, let σ be the seed s-tile and let n be the size of $[\Gamma] - [\sigma]$. Consider one arbitrary derivation of Γ and the graph $G = (V, E)$ constructed in the following way: For $1 \leq i \leq n$ call p_i the position where the i^{th} attachment occurs. Define $V = [\Gamma]$, $E_1 = \{(p_i, p_j) | i < j\}$ and $E_b = \{(p, p') | p \in [\sigma]$ and $p' \in [\Gamma] - [\sigma]$, and p and p' are position-adjacent $\}$. Define a DAG $G' = ([\sigma], E_s)$ such that $E_b \subset E_s$, G' is connected, for all edges (p, p') in E_s p and p' are position adjacent, and G' has a single source. Define $E = E_1 + E_b + E_s$.

To complete the proof, we have to verify that G is an EAG. We start by proving G is a DAG. Because of the way we constructed it, there cannot be a cycle in G'. We observe that all edges in E_1 connect vertices in $[\Gamma] - [\sigma]$, that all edges in E_b go from a vertex in $[\sigma]$ to a vertex in $[\Gamma] - [\sigma]$ and that all edges in E_s connect elements in $[\sigma]$. Therefore, there cannot be a path from a vertex in $[\Gamma] - [\sigma]$ to $[\sigma]$. There are not cycles in G', so if there is a cycle in G it must contain exclusively elements in $[\Gamma] - [\sigma]$. By the definition of E_1 this is clearly impossible, hence G is a DAG. Now we prove that G has exactly one source, i.e. the one in $[\sigma]$. We claim that there cannot be a source in $[\Gamma] - [\sigma]$: Assume there is one and let p_i be that vertex. If there is a vertex $p \in [\sigma]$ such that p_i and p are position-adjacent then, by definition of E_b, p_i has an incoming edge and is not a source. Now consider the case when there is not such a p. Because of the definition of E_1 we know that for all all outgoing edges (p_i, p_j), $i < j$. Therefore, in our derivation the attachment at i happened before the attachment at j. If p_i is a source, it means that the attachment at p_i happened before any of its position-adjacent locations were filled. No derivation can exhibit that behavior.

To verify the last condition of the definition of EAG, we simply observe that E was obtained from an actual derivation. ■

In the previous proof we used a derivation of the s-tile to construct a suitable EAG. We say that the EAG was *induced* by the derivation.

Sentinel process: Let $\mathbf{T} = \langle T, S, g, \tau \rangle$ be a tile system that uniquely produces an s-tile, let Γ be that s-tile, let P be a concentrations function for **T**, let G be a EAG for **T** and let g_s be the source of G. Let S be the set of all subgraphs $G' = (V', E')$ of G such that $g_s \in V'$ and for all edges $(p, p') \in V$, $(p, p') \in E'$ iff $p, p' \in V'$.

We define $S_{\mathbf{T},G,P}$ to be the continuous time Markov process such that:

1. Each state in $S_{\mathbf{T},G,P}$ corresponds to a graph in S.

2. There is a transition from state s' to s'' in $S_{\mathbf{T},G,P}$ iff the corresponding subgraphs $G' = (V', E')$ and $G'' = (V'', E'')$ are such that $V'' - V'$ is a singleton and $E'' - E' = \{(u, v)|v \in V'' - V$ and $u \in Pred_G(v)\}$.

3. The probability associated with a transition from s to s' is $P(v)$, where v is defined as before.

Note that $S_{\mathbf{T},G,P}$ has exactly one source state, corresponding to the source of G, and exactly one sink state, corresponding to G. We define the completion time for the sentinel process as the time to go from the source state to the sink state. Intuitively, the sentinel process is obtained by modifying the self-assembly process by deleting some transitions in it, i.e. *disallowing* some bonds to form. This elimination of transitions will make the completion time of the sentinel process an upper bound for the running time of the self-assembly process.

Completion time for the sentinel process: Consider a tile system **T** that uniquely produces an s-tile, and a concentrations function P for **T**, and let $c = min_i(P(i))$, i.e. the smallest concentration. Let G be an EAG for **T** and let L be the length of the longest directed path in G. Let t be the completion time for $S_{\mathbf{T},G,P}$.

Lemma 4.2 $\mathbf{E}[t] = O(L/c)$. *Further, t has an exponentially decaying tail.*

Proof: Let $P_1, P_2, \ldots P_N$ represent the N directed paths from the seed to any sink in the sentinel graph. At each step in any path, the edge must go to a position-adjacent vertex and there are at most three candidates. Therefore N is at most $3^L \le e^{2L}$. Let S_l denote the sum of all $X_{i,j}$ such that position (i, j) lies on path P_l. Then the completion time $t = \max_{l=1}^{n} S_l$. S_l is the sum of at most L mutually independent exponential variables, each with mean less or equal to $1/c$. Hence $\mathbf{E}[S_l] \le L/c$; let ϕ denote the value L/c. Clearly $\phi = O(L/c)$. Using Chernoff bounds for exponential variables [5], it follows that $\mathbf{Pr}[S_l > \phi \cdot (1 + \delta)] \le ((1 + \delta)/e^{\delta})^L$. Hence $\mathbf{Pr}[t > \phi(1 + \delta)] \le N((1 + \delta)/e^{\delta})^L \le e^{2L}((1 + \delta)/e^{\delta})^L = ((1 + \delta)/e^{\delta-2})^L$. Let us choose $\delta = \delta' + 4$, where $\delta' > 0$. Now

$$\begin{aligned}
&\mathbf{Pr}[t > 5\phi(1 + \delta')] \\
&\le \mathbf{Pr}[t > \phi(1 + 4 + \delta')] \\
&\le ((1 + \delta')/e^{\delta'})^L.
\end{aligned}$$

This clearly gives an exponential tail bound. Now,

$$\mathbf{E}[t] \le 5 \cdot \phi(1 + \int_{\delta'=0}^{\infty} ((1 + \delta')/e^{\delta'})^L d\delta'),$$

which is $O(L/c)$ as $\phi = O(L/c)$ and the integral is bounded by a constant for any value of $L \ge 1$. ■

Stochastic dominance: Define $t_{i,j}$ to be the time at which the (i, j) position gets filled in the original self-assembly process and $t'_{i,j}$ to be the time at which it gets filled in the sentinel process. If (i, j) is in the shape of the seed s-tile then assume $t_{i,j} = 0$. Note that $t_{i,j}$ and $t'_{i,j}$ are random variables. Let t and t' be the random variables denoting the times at which the original self-assembly process and the sentinel assembly complete.

A real valued random variable A is said to be stochastically dominated by another random variable B, denoted $A \leq_{sd} B$, if for all x, $\mathbf{Pr}[A > x] \leq \mathbf{Pr}[B > x]$.

Lemma 4.3 *For all (i, j) in the shape of the uniquely produced s-tile, $t_{i,j} \leq_{sd} t'_{i,j}$*

Proof: Let Γ be the s-tile produced by **T** Let $X_{i,j}$ be an exponential random variable with mean $1/P(\Gamma((i, j)))$, i.e. the reciprocal of the probability associated with the tile at position (i, j) in the final s-tile. Let all the $X_{i,j}$ be independent. A tile attaches at position (i, j) in the self-assembly $X_{i,j}$ time after this position becomes attachable[1]. We couple the sentinel process and the self-assembly process by setting the values of $X_{i,j}$ to be the same for both processes. Define $a_{i,j}$ and $a'_{i,j}$ to be the times at which tile position (i, j) is attachable in the self-assembly process and the sentinel process, respectively. Let t be the earliest time when a tile gets attached in the sentinel process but is still unattached in the self-assembly process. Let (i, j) be this tile position. Clearly, $a'_{i,j} < t$. Therefore any tiles which had attached in the sentinel process by time $a'_{i,j}$ had also attached in the self-assembly process. Since the sentinel process was formed by *disallowing* certain bonds in the self-assembly process, tile position (i, j) is also attachable in the self-assembly process at time $a'_{i,j}$. Hence $a_{i,j} \leq a'_{i,j}$. But $t_{i,j} = a_{i,j} + X_{i,j}$ and $t'_{i,j} = a'_{i,j} + X_{i,j}$. This implies that $t_{i,j} \leq t'_{i,j}$, which is a contradiction. Since $t_{i,j} \leq t'_{i,j}$ for each coupled experiment, $t_{i,j} \leq_{sd} t'_{i,j}$. ■

Lemma 4.3 in conjunction with the lemma 4.2 now allows us to conclude:

Theorem 4.4 $\mathbf{E}[t] = O(L/c)$. *Further, t has an exponentially decaying tail.*

Consider a unit-seed tile system **T** that uniquely produces a s-tile, and a constant valued concentrations function C for **T**. Let c be the value of C.

Theorem 4.5 *There exists an EAG G for* **T** *such that the time complexity of* **T** *is* $\Theta(L/c)$.

Proof: Let Γ be the terminal s-tile of **T**, and let t be the time complexity of **T**. Consider an experiment, i.e. the assembly of Γ from the seed. Let $\tilde{t}$ be the completion time of the experiment.

Define $p_1 = \mathbf{Pr}\left[\tilde{t} \leq 2t\right]$. Clearly, $\mathbf{E}[\tilde{t}] = t$ and by Markov's inequality $p_1 \geq 1/2$.

Let $\tilde{G}$ be the EAG induced by the experiment, that can be constructed as we shown in the proof of Theorem 4.1. Let $\tilde{L}$ be the length of the longest directed path in $\tilde{G}$. Define $p_2 = \mathbf{Pr}\left[\tilde{L}/(4c) \geq \tilde{t}\right]$. We will now show that $p_2 < 1/4$.

Define $\beta = x_1 + x_2 + \cdots + x_{\tilde{L}}$ where the $x_i's$ are independent exponential random variables with mean $1/c$. Using the fact that exponential variables are the inter-arrival times of Poisson events we get:

$\mathbf{Pr}\left[\beta \leq \tilde{L}/(4c)\right] = \mathbf{Pr}\left[\rho(\tilde{L}/4) \geq \tilde{L}\right]$

where $\rho(\tilde{L}/4)$ is a Poisson variable with mean $\tilde{L}/4$. Using Markov inequality again, we obtain $\mathbf{Pr}\left[\rho(\tilde{L}/4) \geq \tilde{L}\right] \leq 1/4p$, hence $\mathbf{Pr}\left[\beta \leq \tilde{L}/(4c)\right] \leq 1/4$. Since $\beta \leq_{sd} \tilde{t}$ we conclude that $\mathbf{Pr}\left[\tilde{L}/(4c) \geq \tilde{t}\right] \leq 1/4$.

[1]This fact follows from the fact that once a tile position becomes attachable it remains attachable till it actually attaches.

Since $p_1 > p_2$, it must be the case that **Pr** $\left[\tilde{L}/(4c) \leq \tilde{t} \wedge \tilde{t} \leq 2t\right] \neq 0$. Therefore, there must exist a derivation that induces an EAG G such that $L/(4c) \leq 2t$ and hence $t = \Omega(L/c)$, where L is the length of the longest directed path in G. From Theorem 4.3 we know that $t = O(L/c)$, completing the proof. ■

For completeness, we state the following lemma of trivial proof that gives a lower bound on the time-complexity of a partial order system. Let **T** be a partial-order system and let G be the DAG describing the partial order relation.

Lemma 4.6 *For all concentrations functions P for* **T**, *the time complexity of* **T** *is* $\Omega(L)$

Theorem 3.5 can be obtained immediately from Lemma 3.4, Lemma 4.6, and Theorem 4.5. We present the details below.

Proof of Theorem 3.5: Let G be the acyclic graph representing the partial order relation in the counter. Lemma 3.4 states that the longest path in G has $\Theta(n)$ length. The $\Theta(n)$ running time follows from Theorem 4.5. A suitable EAG is G, augmented with all edges of the form $((x+1,y),(x,y))$, where both $(x+1,y)$ and (x,y) are in the seed row, and with edges of the form $((x,y),(x,y+1))$, where (x,y)'s are in the seed row, (see Fig 2). It's easy to see that the length of the longest directed path in the augmented graph is still $\Theta(n)$. To prove the time complexity is $\Omega(n)$, we simply invoke Lemma 4.6.

5 Simulating base conversion by self-assembly.

In Rothemund and Winfree's construction (and in ours), the "seed row" represents a number written in binary and consists of $\log n$ tiles, each encoding 0 or 1. The number is used as the starting point for a counter. Since we are only allowed a seed s-tile consisting of a single tile, this seed row itself needs to be assembled from a single tile. Each tile in the seed row needs to be distinct, and therefore the program size complexity for assembling the seed row itself is $\Omega(\log n)$. On the other hand, Kolmogorov complexity dictates the program size complexity to be $\Omega(\frac{\log n}{\log\log n})$. In order to achieve the Kolmogorov bound, we could use a "seed row" to represent a number written base b, where b is power of 2 such that

$$\frac{\log n}{\log\log n} \leq b = 2^k < \frac{2\log n}{\log\log n},$$

for k a positive integer. Then the number of tiles to construct the seed row will be

$$h = \frac{\log n}{\log b} < \frac{\log n}{\log\log n - \log\log\log n} = O(\frac{\log n}{\log\log n}).$$

As in Rothemund and Winfree [8]'s counting scheme, we could use the "seed row" as a starting point for a counter. If we use the counting strategy of Rothemund and Winfree, modified to count base b, we easily achieve the optimal program size complexity or in the language of Rothemund and Winfree:

Theorem 5.1 $\mathbf{K}^2_{SA}(N) = O(\frac{\log N}{\log\log N})$.

Unfortunately if we simply follow Rothemund and Winfree [8]'s method, we pay a price in time complexity. The Rothemund and Winfree construction begins with the seed row and grows through a succession of s-tiles to produce a final rectangle. But for each s-tile produced in the process, there is exactly one tile which can be incorporated. There are at least $b = \Theta(\frac{\log n}{\log\log n})$ distinct tiles which occur $\Omega(n)$ times in the final rectangle, and at least one of these tiles must have probability $O(\frac{\log\log n}{\log n})$. It follows that such tiles require $\Omega(\frac{n\log n}{\log\log n})$ time for assembling of the rectangle. In particular, the time is not linear. Even using the parallel construction described in section 3, the time is not improved.

To overcome this time problem while preserving the $O(\frac{\log n}{\log\log n})$ program size, we adopt the strategy of writing a "seed column" base b and then converting it to base 2, forming a seed row for a counter. We then employ the fast parallel counter described in section 3.

In the conversion part, we will employ some tiles, which have the capability to perform division-by-2. These tiles have glues representing binary strings. The north glue represents last bit, the west glue represents the prefix substring and the east glue represents the string itself. See Figure 5 (A) for the tile associated with the string 0110, and Figure 5 (B) for cooperation of two tiles. We assume that the temperature is 3, but it is not hard to see that the same idea also works for temperature 2.

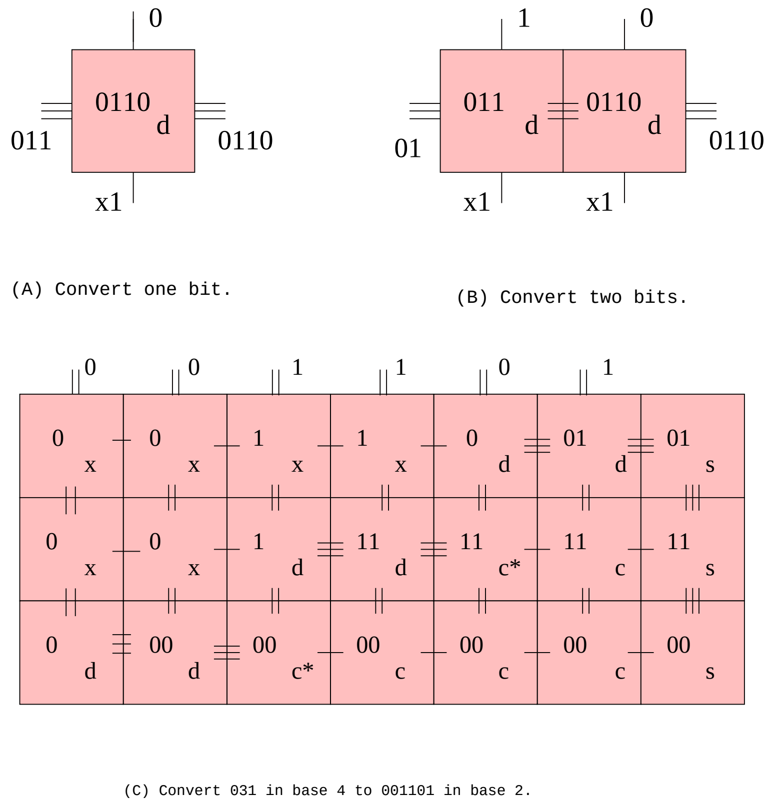

Figure 3: Converting base by self-assembly.

In the following tile descriptions, we identify a tile by the labels on its four sides: North(N), West(W), East(E) and South(S), and a title symbol(TS). In our notations, a prefix "(2)" indicated a glue with strength 2, a prefix "(3)" indicated a glue with strength 3. The absence of prefix indicates glue with strength 1.

Suppose we want to convert $b_1 \cdots b_h$ in base b to binary base number, where b_i if written in binary would be $b_{i,1}b_{i,2}\cdots b_{i,k}$. The set of tiles consists of

1. (Seed.) These tiles will form a seed column. For each $b_{i,1}b_{i,2}\cdots b_{i,k}$, $1 \leq i \leq h$,

   TS: $(b_{i,1}b_{i,2}\cdots b_{i,k})_{\rm s}$
   N: if $i < h$, (3)S_i; S: If $i > 1$, (3)S_{i-1};
   W: if $i = h$, (3)$b_{h,1}b_{h,2}\cdots b_{h,k}$, else $b_{i,1}b_{i,2}\cdots b_{i,k}$.

2. (division by 2.) For each binary string $a_1a_2\cdots a_i$, $1 \leq i \leq k$,

   TS: $(a_1a_2\cdots a_i)_{\rm d}$,
   N: (2)a_i, E: (3)$a_1a_2\cdots a_i$,
   W: if $i \geq 2$, (3)$a_1a_2\cdots a_{i-1}$, otherwise x2,
   S: if $i \geq 2$, (2)x1, otherwise (2)x3.

3. (Base b copy.) For each binary string $a_1a_2\cdots a_k$,

   TS: $(a_1a_2\cdots a_k)_{\rm c}$,
   N: (2)x1, W: $a_1a_2\cdots a_i$, E: $a_1a_2\cdots a_i$, S: (2)x1.

4. (Last b copy tile.) For each binary string $a_1a_2\cdots a_k$,

   TS: $(a_1a_2\cdots a_k)_{\rm C*}$
   N: (2)x3, W: (3)$a_1a_2\cdots a_i$, E: $a_1a_2\cdots a_i$, S: (2)x1.

5. (Base 2 copy.) For $a = 0$ or 1,

   TS: $a_{\rm x}$; N: (2)a; E: x2; S: (2)a; W: x2.

Note that the east side of the seed tiles, the south side of the first seed tile and the north side of the last seed tile are not assigned with any glues. They are open for later use. Even if those glues are all different, the number of distinct tiles is not increased.

Figure 5(C) is an example showing how to convert string 031 in base 4 to string 001101 in base 2.

In item 2, we use the most distinct tiles. which is

$$b + \frac{b}{2} + \frac{b}{4} + \cdots 1 < 2b = O(\frac{\log n}{\log\log n}).$$

Thus the number of distinct tiles in this conversion subroutine is $O(\frac{\log n}{\log\log n})$. Every tile appears at most $O(\log n)$ times. We can arrange that each tile has probability greater than $\Omega(\frac{\log^2 n}{n\log\log n})$, in which case, the time to do the conversion is at most $O(n)$.

6 Putting it all together

The self-assembly of an $n \times n$ square begins with a seed tile which grows into a seed column consisting of roughly $b = \Theta(\frac{\log n}{\log\log n})$ tiles representing a number m written base b. This seed row spawns a base

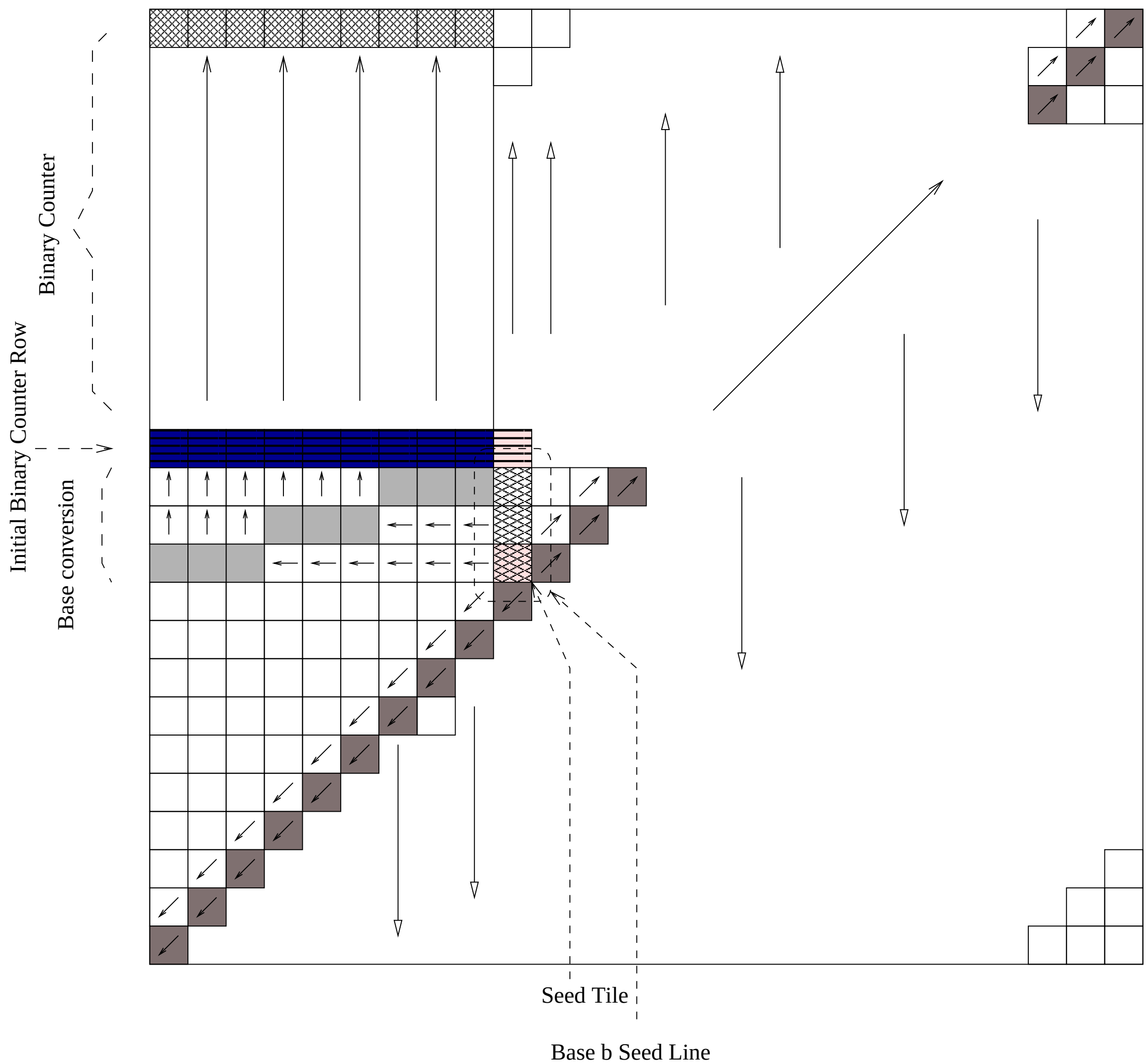

Figure 4: Constructing a square by self-assembly.

conversion assembly as outlined in section 5 The result is a rectangle with top row representing m written base 2. This top row becomes the starting point for a binary counter assembly as outlined in section 3.

The height of the resulting counter rectangle is a function H of m. H satisfies $H(m+1) = H(m) + 3$. Hence by selecting an appropriate m, the base conversion assembly and the counter assembly have dimensions such that height plus width equals $n - e$, where $e = 0, 1$ or 2. We can grow e extra rows at the base of the combined rectangle to bring the sum of the two dimensions exactly up to n. The resulting rectangle acts as the basis for the completion of the $n \times n$ square as described in [8] using diagonal elements and filler tiles. Figure 4 describes this process pictorially.

Let $\mathbf{T}_n$ be the variant of the tile system described above that assembles an $n \times n$ square. Note that only the seed row and base conversion tiles have to be adjusted to specify n. The tiles to build the square, the diagonal and the filler tiles do not depend on n.

Theorem 6.1 *For all positive integers n, there is a concentration functions P_n for $\mathbf{T}_n$ such that $\mathbf{T}_n$ assembles an $n \times n$ square in $O(n)$ time. Further, for all concentrations functions P_n for $\mathbf{T}_n$, the square assembles in $\Omega(n)$ time.*

Proof:

$\mathbf{T}_n$ is a partial-order system. We can use the DAG describing the partial-order relation as EAG. We claim he length of the longest directed path in the DAG is $\Theta(n)$, since by Lemma 3.4 the longest path in the counter is of length $\Theta(n)$, the total number of positions in the seed columns and base conversion is $O(\log^2(n)/\log\log(n))$, and the longest path in the rest of the square is of length $\Theta(n)$. By allocating a combined probability of $1/5$ to the tiles constructing the seed row, $1/5$ to the tiles which perform the base conversion, $1/5$ to the tiles which perform the counter construction, $1/5$ to the diagonal elements and $1/5$ to the filler tiles, it follows from Theorem 4.4 that the time to construct the $n \times n$ square is $O(n)$. By Lemma 4.6 the time complexity is $\Omega(n)$. ■

7 Bounding running time of a self assembly process

In section 3.2 we constructed the equivalent acyclic graph based on the partial order relation existing among the times to fill every position of the counter. The technique is not always applicable. There are self-assembly processes that do not exhibit the partial order relation. We present in this section a different parallel counter that is not a partial order system and we will show how to bound the time to complete its assembly.

The fully assembled counter is going to be a rectangle, with K tiles in each row. Each row represents a number between 0 and n – this number can be obtained by reading the main labels on the tiles in the row, with the most significant bit being the west-most in the row. We describe the self-assembly process which results in the counter being incremented.

The increment operation happens in three stages and uses 15 different tiles (see figure 5). We start with an "INERT" row (each tile carries the auxiliary label "I"). This row gets replicated into an "ACTIVE" row (auxiliary label "A") if and only if there is at least one 0-tile in the original INERT row. If all the tiles in the INERT row are 1-tiles, the counter construction can not proceed any further and the self-assembly process terminates. A "CARRY" row (auxiliary label "C") then assembles on top of the ACTIVE row.

The bulk of the increment operation happens during this step. The CARRY row will represent the value of the ACTIVE row incremented by one, except that the 0-tile (if any) which needs to change to a 1-tile due to a carry propagation is still unchanged. Then an INERT row assembles on top of the CARRY row which will convert such a tile (if any) to a 1-tile. The east-most tile in any row always carries the auxiliary label "R" along with the auxiliary label corresponding to the row. Some tiles may also be marked special (auxiliary label "S") to aid in the carry propagation. Even though the above description seems serial, the SA-counter need not assemble row-by-row; several different tiles (possibly belonging to different rows) may be attachable at the same time.

We will assume that the temperature for this counter construction is 3. We now describe the tiles in each of the rows and their glues. All tiles occur with the same probability, p_{SA} which is a constant independent of n.

INERT tiles: There are the following INERT tiles: 0L, 1L, 0LE, 1LE, 0LL, 1LL.

ACTIVE tiles: 0L, 1L, 0LE, 1LE.

CARRY tiles: 0L, 1L, 0LL, 0LE, 1LE. The glues and their strengths are depicted pictorially in figure 5.

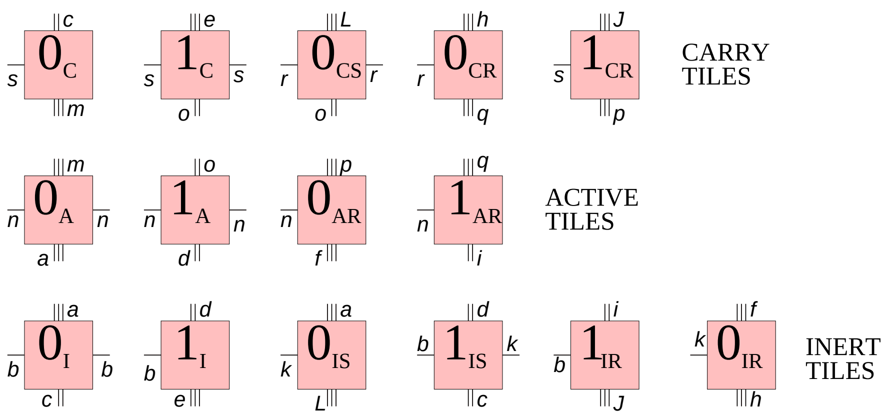

Figure 5: The tiles for the SA-counter. The number of lines jutting from each edge of the tile represent the strength of the bond (1,2, or 3) and the label on the edges represents the glue. Some edges do not have any glues.

Assembling ACTIVE rows: Notice that starting with an INERT row that consists of all 1-tiles, no ACTIVE tiles can attach to the top of the row since all possible bonds are of strength 2 whereas the temperature is 3. However, all 0-tiles in the INERT row can allow ACTIVE 0-tiles to attach on top through bonds of strength 3. These newly attached ACTIVE 0-tiles provide strength 1 bonds that supplement the strength 2 bonds between ACTIVE 1-tiles and INERT 1-tiles, allowing ACTIVE 1-tiles to attach in positions adjacent to the already attached ACTIVE 0-tiles. The newly attached ACTIVE 1-tiles in turn provide strength 1 bonds that allow adjacent ACTIVE 1-tiles to attach on top of INERT 1-tiles, and so on. This allows the entire INERT row to be replicated into an ACTIVE row.

Assembling CARRY rows: Notice that any 0-tiles (except the east-most) in the ACTIVE row allow a CARRY 0-tile to attach on top by means of bonds of strength 3. These tiles then allow CARRY

1-tiles to attach on the west through bonds of strength 1, which in turn allow more CARRY 1-tiles to attach on the west. Notice that CARRY 0-tiles do not provide any glues on the east and therefore can not facilitate attachment of CARRY 1-tiles to their east. This allows most of the ACTIVE row to replicate into the CARRY row – we just need to bother about the east-most tile and those ACTIVE 1-tiles that do not have an ordinary ACTIVE 0-tile to their east. We will now consider two cases depending on the east-most tile in the ACTIVE row. If the east-most tile in the ACTIVE row is a 0-tile then it will allow a east-most CARRY 1-tile to attach on its top through a bond of strength 3. This then provides a bond of strength 1 to its west which will allow ACTIVE 1-tiles to assemble to its west, thus completing the CARRY row. Notice that this CARRY row already represents the correct result for the increment operation. If the east-most tile is an ACTIVE 1-tile, then it allows a east-most CARRY 0-tile to attach on top. This tile then allows special CARRY 0-tiles (and not CARRY 1-tiles) to bond to its west on top of the ACTIVE 1-tiles. Thus the entire sequence of contiguous ACTIVE 1-tiles on the east will have special CARRY 0-tiles attached on top. The CARRY row now represents the correct result for the CARRY operation with one exception: the ACTIVE 0-tile which was immediately to the west of the east-most sequence of ACTIVE 1-tiles has a CARRY 0-tile attached on top, whereas the increment operation should convert it into a 1-tile due to carry propagation. This is remedied in the next step.

Assembling INERT rows: The 0-tile which needs to be changed into a 1-tile can be "detected" as a CARRY 0-tile which has a special CARRY 0-tile or a east-most CARRY 0-tile to its immediate east. The description for the assembly of the INERT row is simple. All the tiles except ordinary CARRY 0-tiles can attach an INERT tile (with the other labels such as the bit-label, the east-most label, and the special label remaining the same) on top. Now special INERT 0-tiles and east-most INERT 0-tiles allow a special INERT 1-tile to attach (on top of a CARRY 0-tile) to their west, while all other INERT tiles allow an INERT 0-tile to attach. This accomplishes the desired carry propagation and completes the increment process.

The Seed Row: The seed s-tile S_K is a row of K special tiles. The east-most tile in S_K is identical to the tile 0_{LE} except that the bottom surface has no glue. The other $K-1$ tiles are identical to the tile 0_{L} except that there is no glue on the bottom surfaces[2].

The tile system $\mathbf{T}_{SA}(K)$ has a tile set consisting of all the tiles in figure 5 as well as the special tiles needed in the s-tile S_K. The glue strength function is as indicated in figure 5. The temperature is 3, and there is a single seed s-tile S_K. We will refer to the Markov chain defined by the tile system as the SA-counter. We will assume that all tiles which are not in S_K have the same constant probability.

The next theorem follows from our description of the tile system, and we omit the proof.

Theorem 7.1 *The tile system $\mathbf{T}_{SA}(K)$ uniquely produces an s-tile which is a $K \times (3 \cdot 2^K - 2)$ rectangle.*

Minor modification of the seed row results in a tile system that uniquely produces an s-tile which is a $K \times n$ rectangle for any $n \leq 3 \cdot 2^K - 2$.

[2] In this section we are using s-tiles consisting of more than one tile as the seed, whereas our restricted model allows us to use only s-tiles with a single tile as seed. We rectify this in section 6.

The equivalent acyclic graph: Look at a completed self assembly, and replace each bond by two directed bonds, one in each direction. Each directed bond has the same strength as the original bidirected bond. Then, remove all the directed bonds that satisfy any of the following criteria:

1. The bond goes from a higher to a lower row.
2. The bond goes from west to east in a non-ACTIVE row.
3. The bond goes from west to east in an ACTIVE row, and there is at least one zero-tile to the east of the origin of the bond.
4. The bond goes from east to west in an ACTIVE row, and the destination tile of the bond is a zero-tile.
5. The bond goes from east to west in an ACTIVE row, the source and destination tiles are both one-tiles, and there are no zero-tiles to the east of the source.

We can define an equivalent graph as the graph with vertices corresponding to the tiles, and directed arcs corresponding to the directed bonds and labeled by the strength. The following lemma is immediate from our construction.

Lemma 7.2 *The sentinel graph is acyclic.*

The sentinel process is a Markov chain obtained by modifying the Markov chain corresponding to the SA-counter as follows. We consider each transition in the Markov chain for the SA-counter. Let this transition correspond to adding a new tile X to an s-tile A. A tile Y in A is said to be a *support tile* if it shares a side with X and there is an arc from Y to X in the sentinel graph; the strength of this arc is said to be the *support strength* from Y. We retain this transition if the sum of the support strengths from the support tiles is greater than the temperature and else we discard it. Any state in the Markov chain that is unreachable from the source state is discarded.

Intuitively, the sentinel process is formed by taking the SA-counter and introducing a "sentinel" who disallows transitions that *require* bond-formation from the west to the east, except when such transitions are necessary for replication. It is not difficult to see that the sentinel process produces exactly the same complete assembly as the SA-counter.

In the sentinel graph, there is exactly one bond that goes from the west-most column to the east, and this happens just above the INERT row $011\ldots11$ where the west-most 0 is the only tile that can "self-replicate" into the active row and must then induce the 1-tiles to attach to its east. Further, this property is recursively satisfied if we remove the west-most column and look at the sentinel graphs for the two sub-rectangles below and above the $011\ldots11$ row. This recursive structure makes the sentinel process more amenable to analysis. The structure of the sentinel graph is illustrated in figure 6.

8 Conclusions and open problems

We have generalized the Tile Assembly Model of Rothemund and Winfree [8] to include time complexity and have demonstrated a tile system for self-assembling $n \times n$ squares that is optimal in both time and program size. The long term goal of this research is the development of a mathematical-computational theory

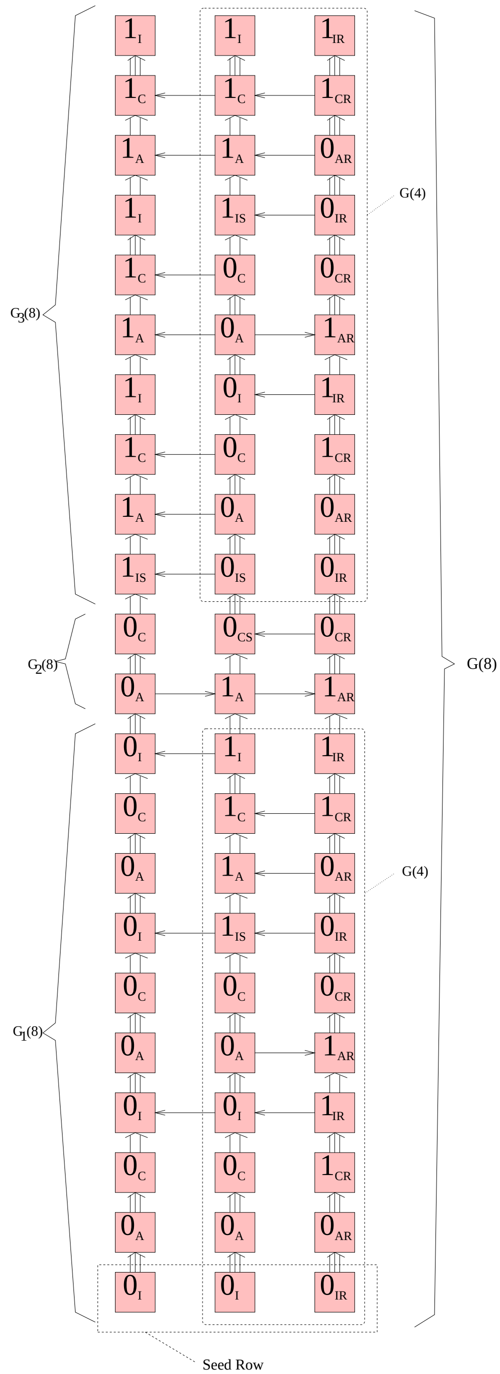

Figure 6: The recursive structure of the sentinel graph.

of self-assembly that will combine classical theories such as thermodynamics and statistical mechanics with modern theories such as combinatorics and computational complexity.

There are many open problems. For example:

1. In the model described here, s-tiles grow by accretion i.e. by the addition of single tiles to a growing crystal. Modify the model to allow any two s-tiles to merge into a larger s-tile. In such a system the lower bound of $\Omega(n)$ for the time to assemble $n \times n$ squares no longer appears necessary. In this case, what is the best possible time and can it be achieved simultaneously with optimal program size?

2. The Tile Assembly Model as presented here is ”irreversible” - once a tile sticks, it never ”unsticks”. Generalize to a model that allows tiles to both attach to and detach from the assembly.

Acknowledgements

We would like to thank Paul Rothemund and Erik Winfree for several helpful discussions about their work [8].

References

[1] L. M. 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, M. Huang, D. Kempe, P. Moisset de Espanés, P. Rothemund: Combinatorial optimization problems in self-assembly. ACM Symposium on Theory of Computing (STOC) 2002. pages 23-32.

[3] H. Abelson, D. Allen, D. Coore, C. Hanson, G. Homsy, T. Knight, R. Nagpal, E. Rauch, G. Sussman and R. Weiss. Amorphous computing. *AI Memo 1665*, August 1999.

[4] C. Mao, T. H. LaBean, J. H. Reif and N. C. Seeman. Logical computation using algorithmic self-assembly of DNA triple-crossover molecules. *Nature*, 407:493–496, 2000.

[5] R. Motwani and P. Raghavan. Randomized Algorithms. *Cambridge University Press, 1995.*

[6] P. Rothemund. Using lateral capillary forces to compute by self-assembly. *Proceedings of the National Academy of Sciences*, 97:984–989, 2000.

[7] P. Rothemund. Theory and Experiments in Algorithmic Self-Assembly University of Southern California Ph.D. Thesis Copyright 2001.

[8] P. Rothemund and E. Winfree. The program-size complexity of self-assembled squares. *STOC 2000*.

[9] H. Wang. Proving theorems by pattern recognition. II. *Bell Systems Technical Journal*, 40:1-42, 1961.

[10] E. Winfree, F. Liu, L. A. Wenzler, and N. C. Seeman. Design and self-assembly of two-dimensional DNA crystals. *Nature*, 394:539-544, 1998.

[11] E. Winfree. PhD thesis. Cal Tech, 1998.

[12] E. Winfree. *Personal communication.*