Debdas Ghosh · Debasis Giri
Ram N. Mohapatra · Ekrem Savas
Kouichi Sakurai · L. P. Singh (Eds.)

Communications in Computer and Information Science 834

Mathematics and Computing

4th International Conference, ICMC 2018
Varanasi, India, January 9–11, 2018
Revised Selected Papers

Springer

Editors
Debdas Ghosh
Department of Mathematical Sciences
Indian Institute of Technology BHU
Varanasi, Uttar Pradesh
India

Debasis Giri
Haldia Institute of Technology
Haldia
India

Ram N. Mohapatra
University of Central Florida
Orlando, FL
USA

Ekrem Savas
Istanbul Commerce University
Istanbul
Turkey

Kouichi Sakurai
Kyushu University
Fukuoka
Japan

L. P. Singh
Indian Institute of Technology (BHU)
Varanasi
India

ISSN 1865-0929 ISSN 1865-0937 (electronic)
Communications in Computer and Information Science
ISBN 978-981-13-0022-6 ISBN 978-981-13-0023-3 (eBook)
https://doi.org/10.1007/978-981-13-0023-3

Library of Congress Control Number: 2018940140

Printed on acid-free paper

This Springer imprint is published by the registered company Springer Nature Singapore Pte Ltd.
part of Springer Nature
The registered company address is: 152 Beach Road, #21-01/04 Gateway East, Singapore 189721,
Singapore

A Study on Complexity Measure of Diamond Tile Self-assembly System

M. Nithya Kalyani, P. Helen Chandra(✉), and S. M. Saroja T. Kalavathy

Jayaraj Annapackiam College for Women (Autonomous),
Periyakulam, Theni District, Tamilnadu, India
rnithraj@gmail.com, chanrajac@yahoo.com, kalaoliver@gmail.com

Abstract. Molecular self-assembly gives rise to a great diversity of complex forms from crystals and *DNA* helices to microtubules and holoenzymes. We study a formal self-assembly model called the Diamond Tile Assembly System in which a diamond tile may be added to the growing object when the total interaction strength with its neighbours exceeds a parameter $\mathcal{T}$. Self-assembled objects can also be studied from the point of view of computational complexity. Here, we define the program-size complexity of an $N \times N$ diamond to be the minimum number of distinct tiles required to self-assemble the diamond. We study this complexity under the Diamond Tile Assembly Model and find a dramatic decrease in complexity from N^2 tiles to $O(logN)$ tiles, as $\mathcal{T}$ is increased from 1 where bonding is non co-operative to 2 allowing co-operative bonding. Further, we observe that the size of the largest diamond uniquely produced by a set of n tiles grows faster than any computable function.

Keywords: Self-assembly · Diamond Tile Assembly
Program-size complexity

1 Introduction

Self-assembly is the process by which a collection of relatively simple components, beginning in a disorganized state, spontaneously and without external guidance coalesce to form more complex structures. The process is guided by only local interactions between the components, which typically follow a basic set of rules. Despite the seemingly simplistic nature of self-assembly, its power can be harnessed to form structures of incredible complexity and intricacy. In order to model such systems, theoretical models have been developed and one of the most popular among these is the Tile Assembly Model introduced by Erik Winfree in his Ph.D. thesis [Wi2]. The complexity of self-assembled shapes is investigated in [LL1,SE1,Su1].

Branched *DNA* molecules [Se1] provide a direct physical motivation for the Tile Assembly Model. *DNA* double-crossover molecules, each bearing four *sticky ends* analogous to the four sides of a Wang tile, have been designed to self-assemble into a periodic two dimensional lattice [WL1]. The binding interactions between double-crossover molecules may be redesigned by changing the base

D. Ghosh et al. (Eds.): ICMC 2018, CCIS 834, pp. 194–204, 2018.
https://doi.org/10.1007/978-981-13-0023-3_19

sequence of their sticky ends, thus allowing arbitrary sets of molecular Wang tiles. From a physically-based stochastic model of such a system, the Tile Assembly Model is obtained in the limit of strong binding domains and low monomer concentrations [Ra1, Wi1]. This model is an extension of the theory of Wang tiles [Wa1] to include a specific mechanism for growth based on the physics of molecular self-assembly.

A *program* consists of a finite set of unit diamond tiles with sides having molecular binding domain and thus each side has an associated *binding strength*, which in our model must be an integer. Starting from a chosen seed tile, growth occurs by addition of single tiles. Tiles bind a growing assembly only if their binding interactions are of sufficient strength as determined by the *temperature* parameter T. T measures the *co-operativity* of the binding interactions. It is interesting to observe that cooperative effects play a major role in gene regulation and many other biological systems.

In this paper, we introduce a new model called Diamond Tile Self-assembly System. It is a formal model for the self-assembly of molecules, such as protein or DNA, constrained to self-assembly on a diamond lattice. We measure the complexity of self-assembly by considering diamond instead of square [RE1]. Standard complexity measures in computer science are based on time, space, program size and decidability. Here, we discuss the program-size complexity of self-assembled diamonds, where complexity is measured by the number of distinct tile types involved.

2 Diamond Tile Self-assembly System

In this section, we introduce a new model called Diamond Tile Self-assembly System.

Definition 1. *A **Diamond Tile Self-assembly System** D_{TAS} is defined by the quadruple*

$$\mathbb{T} = <T, S, g, \mathcal{T}>$$

where T is a finite set of diamond tile types containing empty, S is a seed assembly with finite domain, g is a strength function and $\mathcal{T} \geq 0$ is the temperature. We consider only $|S| = 1$, where $S = A_s^{(0,0)}$.

*Diamond tile self-assembly is defined by a relation between configurations: $A \rightarrow_{\mathbf{T}} B$ if there exists a diamond tile $t \in T$ and a site (x, y) such that $B = A + A_t^{(x,y)}$ and B is $\mathcal{T}$-stable. In particular, at $\mathcal{T} = 1$, a diamond tile may be added if it makes any bond to a neighbour, whereas at $\mathcal{T} = 2$, the diamond tile to be added must either make two weak bonds or a single strong bond. $\rightarrow_{\mathbf{T}}^*$ is the reflexive and transitive closure of $\rightarrow_{\mathbf{T}}$. The diamond tile self-assembly system defines a partially ordered set, the **produced** assemblies $D_{Prod(\mathbf{T})}$ where*

$$D_{Prod(\mathbf{T})} = \{A, \exists S \in T \, s.t. \, S \rightarrow_{\mathbf{T}}^* A\} \, and \, A \leq B \, iff \, A \rightarrow_{\mathbf{T}}^* B.$$

*Another set, the **terminal** assemblies $D_{term(\mathbf{T})}$ is defined as the maximal elements of $D_{Prod(\mathbf{T})}$:*

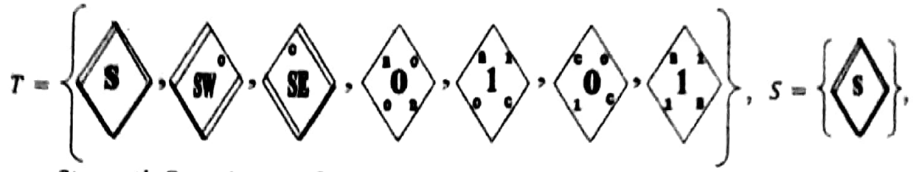
$$D_{Term(\mathbb{T})} = \{A \in D_{Prod(\mathbb{T})}, \nexists B \; s.t. \; A < B\}.$$

The produced assemblies include intermediate products of the self-assembly process, whereas the terminal assemblies are just the end products and may be considered as the output. If

$$A \in D_{Prod(\mathbb{T})} \Rightarrow \exists B \in D_{Term(\mathbb{T})} \; s.t. \; A \rightarrow_{\mathbb{T}}^{*} B$$

then $\mathbb{T}$ *is said to be* ***haltable****, in the sense that every path of self-assembly can eventually terminate. If* $\mathbb{T}$ *is haltable and* $D_{Term(\mathbb{T})}$ *is finite,* $\mathbb{T}$ *is said to be* ***halting*** *in the sense that every path of self-assembly does eventually terminate. In general, if* $D_{Prod(\mathbb{T})}$ *is a lattice, we say that* $\mathbb{T}$ *produces a* ***unique pattern****-*$\mathbb{T}$ *need not be halting nor even haltable.*

Example 1. Consider the Diamond Tile Self-assembly System $\mathbb{T} = <T, S, g, \mathcal{T}>$ where



g = *Strength Function*, $\mathcal{T}$= 2.

The tile set T, consists of four diamond rule tiles with strength-1 binding domains, two border diamond tiles with strength-1 and 2 binding domains and one seed diamond tile with strength-2 binding domains. At $\mathcal{T} = 2$, these tiles

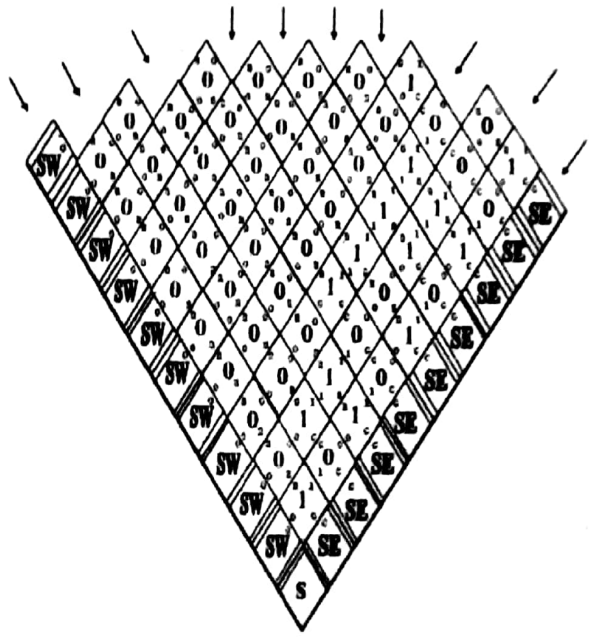


Fig. 1. Simulating a binary counter with diamond self-assembly

count in binary; the n^{th} row above the origin represents the integer n (which is rotated 45° anticlockwise). This self-assembly *program* is analogous to an infinite loop-there are no terminal assemblies. Diamond rule tiles may be added only if both their southwest and southeast neighbors are already in place and there is a unique diamond rule tile for each possible pair of binding domains the neighbors could present; furthermore, the property that only northwest and northeast sides exposed in the assembly is preserved from step to step. The computation is possible when the system temperature = 2 and at least two strength −1 bonds must cooperate for a tile to be added to an assembly. The assembly is not terminal and arrows indicate positions at which may grow. The picture pattern generated by D_{TAS} is shown in Fig. 1.

3 Complexity of Diamond Self-assembly

In this section, we introduce Complexity of Diamond Self-Assembly. We will be measuring program-size complexity using asymptotic notion.

All functions will be from $\mathbb{N} \rightarrow \mathbb{N}$. A function $f(n)$ is **non-decreasing** iff $\forall\, n$, $f(n) \leq f(n+1)$. A function $f(n)$ is **un bounded** iff $\forall\, c, \exists\, n$ s.t. $f(n) \geq c$. We say $f(n) = O(g(n))$ iff $\exists\, c, n_0$ s.t. $\forall\, n > n_0$, $f(n) \leq cg(n)$. We say $f(n) = \Omega(g(n))$ iff $\exists\, c, n_0$ s.t. $\forall\, n > n_0$, $f(n) \geq cg(n)$. We assert proposition $P(n)$ **infinitely often** iff $\forall\, n_0 > 0$, $\exists\, n > n_0$ s.t. $P(n)$. Define $O_{i.o.}$ (big-O infinitely often) such that $f(n) = O_{i.o.}(g(n))$ iff $\exists\, c$ s.t. $f(n) \leq cg(n)$ infinitely often. We assert proposition $P(n)$ **for almost all** n iff $lim_{n_0 \rightarrow \infty} \frac{|1 \leq n \leq n_0 s.t. P(n)|}{n_0} = 1$. Define $\Omega_{a.a}$ (big-Ω almost always) such that $f(n) = \Omega_{a.a}(g(n))$ iff $\exists\, c$ s.t. $f(n) \geq cg(n)$ for almost all n.

We can now formally describe the program-size complexity of an $N \times N$ diamond. An assembly A is an $N \times N$ **diamond** if there exists a site (x_0, y_0) such that $(x, y) \in A$ iff $x \geq x_0$ and $x < x_0 + N$ and $y \geq y_0$ and $y < y_0 + N$. In other words the choice of tiles may be arbitrary, so long as they are there. Diamond A is a **full diamond** if for all (x, y) and $(x', y') \in A$ such that (x, y) and (x', y') are neighbours (x, y) and (x', y') bind with non-zero strength. In other words, every adjacent pair of tiles must have non-zero interaction strength. We are interested in which diamonds can be self-assembled by tile systems:

$$\begin{aligned} D^{\mathcal{T}} = \{&(N, n) \in \mathbb{N} \times \mathbb{N} \text{ s.t. there exists a tile system} \\ &\mathbb{T} = <T, S, g, \mathcal{T}>, |T| = n + 1 \text{ and } \mathbb{T} \\ &\text{uniquely produces an } N \times N \text{ full diamond}\}. \end{aligned}$$

We define the program size complexity $K^{\mathcal{T}}_{DA}(N)$ of a diamond to be the minimum number of distinct non-empty tiles required to uniquely produce the diamond-physically the number of distinct types of molecules that must be prepared.

$$K^{\mathcal{T}}_{DA}(N) = min\{n\, s.t. (N, n) \in D^{\mathcal{T}}\}$$

Our investigations rely on several constructions. We need an easy way to verify that these constructions do indeed *uniquely* produce the target structure. For

each construction, the argument is an elaboration of the argument given for the binary counter tiles, only now an assembly may have more than one diagonal growth front. Specifically, the property that is preserved from step to step is that the assembly is *stop-sign*-shaped: the orientations of the exposed sides along the (clockwise) perimeter are of the form

$$NE^*\{NE, SE\}^*SE^*\{SE, SW\}^*SW^*\{SW, NW\}^*NW^*\{NW, NE\}^*.$$

These arguments rely on showing that there is exactly one strength-2 bond joining each row and each column.

We begin by studying $\mathcal{K}^{\mathcal{T}}_{DA}(N)$ for $\mathcal{T} = 1$ and obtain the following theorem.

Theorem 1. $\mathcal{K}^1_{DA}(N) = N^2$.

Proof. *To show $\mathcal{K}^1_{DA}(N) \leq N^2$, we construct N^2 diamond tiles, one for each position in the diamond, with a unique strength-1 binding domain for each adjacent pair of diamond tiles as in Fig. 2. In Fig. 2 (a): $N^2 = 16$ tiles with unique side labels uniquely produce a terminal 4×4 full diamond at $\mathcal{T} = 1$. (b): $2N - 1 = 7$ tiles uniquely produce a 4×4 diamond (but this is not a full diamond since thick sides have strength 0). Except for the sides labels with a circle, each interacting pair of tiles share a unique side label. This construction is conjectured to be minimal for diamonds assembled at $\mathcal{T} = 1$.*

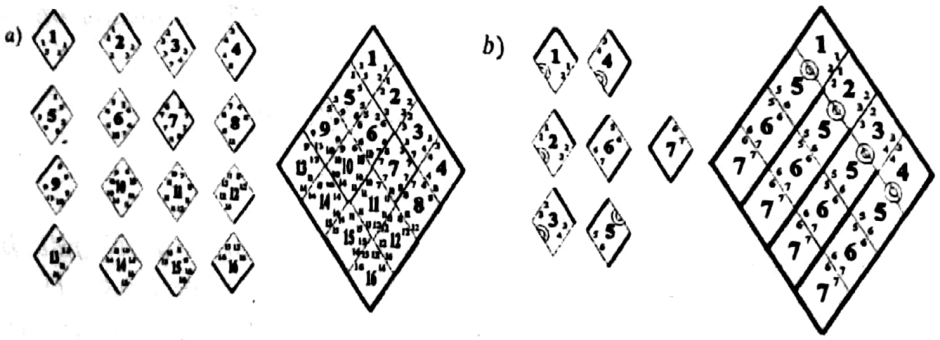


Fig. 2. Formation of diamonds at $\mathcal{T} = 1$.

To show $\mathcal{K}^1_{DA}(N) \geq N^2$, suppose a diamond tile set T with $|T| < N^2$ produces an $N \times N$ full diamond A (Fig. 3). In Fig. 3, a full $N \times N$ diamond with fewer than N^2 diamond tiles must have some tile i present at two sites. Consider the assembly W (the white diamond tiles) which includes an assembly V (bounded diamond tile i), the seed tile S and a tile that connects the seed tile to V. W can be extended indefinitely with the addition of translated segments of V (e.g. V^2_{+1} shown in gray). Then some tile i is present at two sites in A, say (x_1, y_1) and (x_2, y_2).

Let V be the V-shaped (or possibly linear) assembly consisting only of the tiles at $(x_1, y_1), \ldots, (x_2, y_2)$; let V^1 be the assembly such that $V^1 + (x_2, y_2) = V$; let V^2 be the assembly such that $V^2 + (x_1, y_1) = V$; let $V^k_n = [V^k(x + n * (x_2 -$

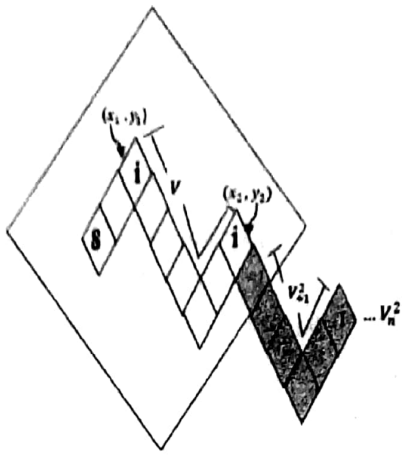


Fig. 3. No $\mathcal{T} = 1$ tile system with fewer than N^2 diamond tiles can uniquely produce an $N \times N$ diamond.

$x_1), y + n * (y_2 - y_1))$ *be a translated version of* V^k *for* $k = 1, 2$ *and let* W *consist of* V, S *and the fewest diamond tiles in* A *required to connect* S *to* V. *Because* W *is contained in* A *and* A *is a full diamond, all adjacent pairs of diamond tiles interact on a strength-(at least)-1 side and therefore* $S \rightarrow_{\mathbb{T}}^{*} W$. *At least one of* $\{V^1_{-1}, V^1_{+1}, V^2_{-1}, V^2_{+1}\}$, *say* V^r_s, *can be added to* W, *resulting in a larger assembly also produced by* $\mathbb{T}$. *This can be continued indefinitely: if* $s = +1$ *then for all* n, $W + \Sigma_{i=+1}^{n} V^r_i$ *is in* $D_{Prod(\mathbb{T})}$; *if* $s = -1$ *then for all* n, $W + \Sigma_{i=-n}^{-1} V^r_i$ *is in* $D_{Prod(\mathbb{T})}$. *This contradicts the assumption that* $\mathbb{T}$ *is halting and terminates in* $N \times N$ *full diamonds.* •

At $\mathcal{T} = 2$ *the situation is markedly different.*

Theorem 2. $\mathcal{K}^2_{DA}(N) = O(N)$.

Proof. *Figure 4 shows two constructions for an* $N \times N$ *full diamond using* $2N$ *(Fig. 4a) and* $N + 4$ *(Fig. 4b) diamond tiles respectively. Diamond tile self-assembly from the seed diamond tile A expands initially by single strength-2 interactions creating the northeast and northwest borders with the alphabetic diamond tiles. As the border grows, two cooperative strength-1 interactions allow the blank tile to fill in and complete the diamond.*

In Fig. 4b, diamond tile self-assembly from the seed diamond tile A expands initially by single strength-2 interactions creating the northeast border with the numbered diamond tiles. The U and V diamond tiles proceeds in the diagonal sides from west to east by their strength-2 interactions. Thus allowing the rest of the column to be filled with blanks. The $N \times N$ *full diamond can be easily verified to be a terminal assembly.* •

This is only the beginning. The construction in Fig. 4b can be combined with a fixed-width version of the binary counter of Fig. 1 to obtain a set of tiles that produce the full diamond by counting in binary instead of by counting in unary.

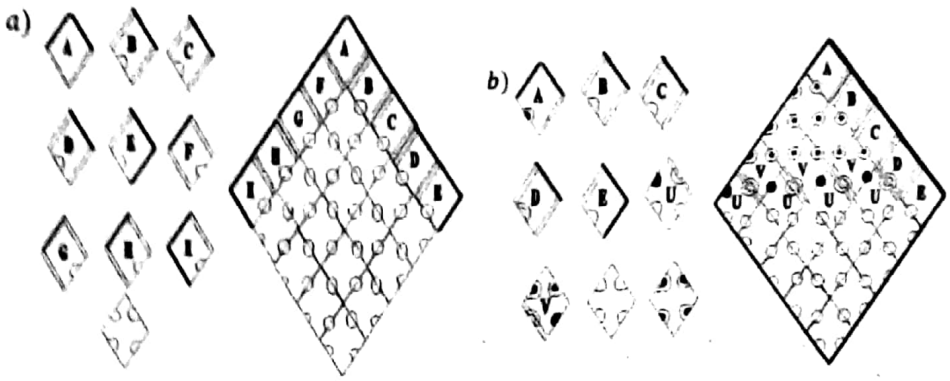


Fig. 4. Formation of full diamonds at $T = 2$.

Theorem 3. $\mathcal{K}^2_{DA}(N) = O(logN)$.

Proof. *Figure 5 constructs an $N \times N$ full diamond using $n+22$ diamond tiles, where $n = \lceil logN \rceil$. $n+2$ diamond tiles, including the diamond seed tile, produce an $(n-1) \times (n-1)$ diamond as in the previous construct (Fig. 4b). Here $N = 52, n = 6$ and 28 tiles are used. Additionally, the $n-1$ diamond tiles in the seed row have northwest sides encoding the bits of the integer $c = 1 + 2^{n-1} - (N - n)/2$, the initial value of the counter. We must use a fixed-width version of the counter diamond tiles of Fig. 1; this requires a special set of diamond tiles for the southwest and southeast columns of bits. The counter counts from c to 2^{n-1} using two rows for each integer. In order to detect when the counter has finished, we use alternating rows to increment the counter from southeast to northwest then to copy of the bits from northwest to southeast unless the northwest bit just rolled over to northeast from 1 to 0. In the latter case, the diamond tile presents a strength-2 side with a label not found on any other diamond tiles, thus halting the counter. (The strength-2 side will be used in our next construction; here, any strength would suffice). There is a special diamond tile for the rightmost bit in the first increment row right the seed row. This diamond tile contains a strength-2 side to initiate the $\mathbf{u} - \mathbf{v}$ diagonal, thus filling in the rest of the diamond. Overall, the counter requires 17 tiles; the seed row requires $n-1$ tiles; the two diagonals require 4 tiles and there are two blank tiles.*

We can do much better: by recursively iterating the above construction one can produce $N \times N$ diamonds with $N \geq \underbrace{2^{2^{2^{\cdots^2}}}}_{\text{ntimes}} \stackrel{def}{=} 2 * * n$. Define $log^* N$ as the least n such that $2 * * n \geq N$.

Theorem 4. $\mathcal{K}^2_{DA}(N) = O_{i.0.}(log^* N)$.

Proof. *Our proof is by induction. Let S^n refer to a diamond tile system containing fewer than $22n$ diamond tiles (including the $\mathbf{u}, \mathbf{v}$ and blank diamond tiles) that uniquely produces an $N \times N$ full diamond such that*

- *$N > 2 * * n$.*
- *All binding domains on the northwest and northwest bottom are strength 1 or 0.*

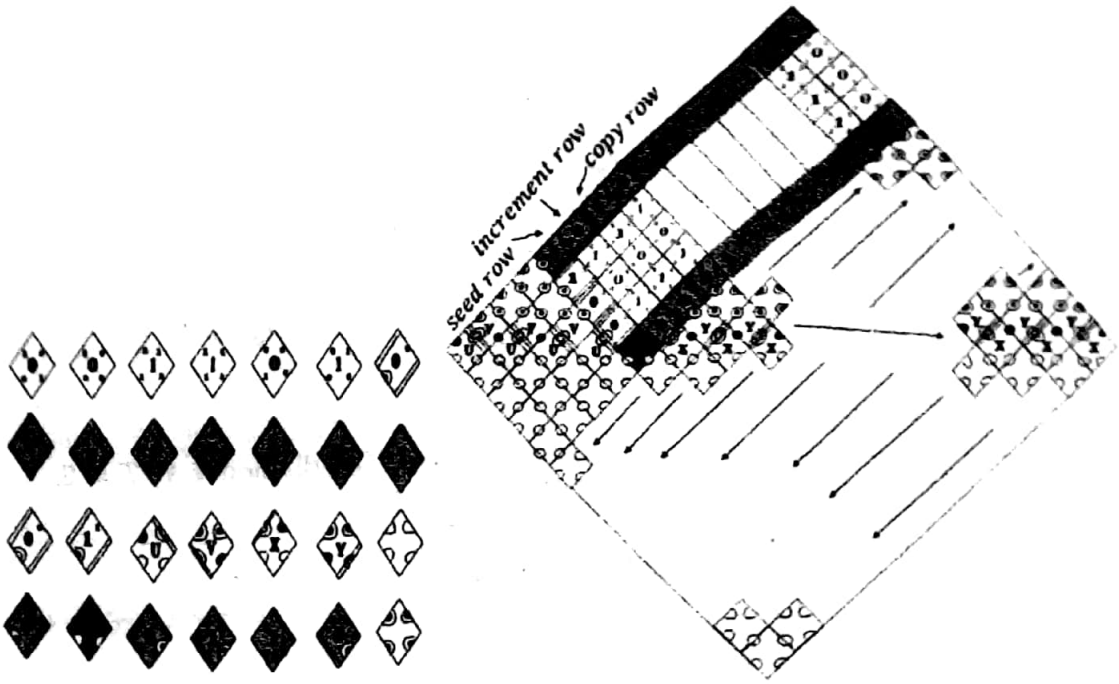


Fig. 5. Formation of an $N \times N$ diamond using $O(\log N)$ tiles.

- *All binding domains on the southwest have the strength-1 blank label.*
- *The binding domains on the northwest upper side conform to the pattern xy^*zb^*a where x is a strength-2 binding domain that occurs nowhere else and y, z, b and a are distinct strength-1 binding domains.*

We show that S^n exists for all n. The base case $n = 1$ is trivial. The inductive step is illustrated in Fig. 6. (In Fig. 6, given a set of tiles S^{n+1} that produce an $N \times N$ full diamond that satisfies recurrence, the addition of 22 new tiles results in S^{n+1} and produces a $(N + 2 \times 2^N) \times (N + 2 \times 2^N)$ full diamond. New side labels (with doubled symbols) prevent counter tiles from S^n from incorporating in the S^{n+1} counter). First, there are 5 diamond tiles that, initiated by x, produce an initial string of $0's$ for a new fixed-width counter and provide a strength-2 side for a new $\mathbf{u} - \mathbf{v}$ diagonal. Then there are 16 diamond tiles equivalent to the counter diamond tile in Theorem 3 but using new side labels; the counter counts to 2^N. The diagonal fills in the rest of the diamond, now with sides of length $N + 2 \times 2^N > 2^N > 2 * * (n+1)$. Therefore S^n exist for all n and for those n,

$$22 \log^* N \geq 22n \geq D^2_{DA}(N).$$

- *$\log^* N$ is an exceedingly slowly growing function; the above construction shows that very large diamonds can be assembled with a very small number of diamond tiles. But we can do much better yet! By embedding the simulation of a Turing Machine in the growth of a diamond we show that:*

Theorem 5. *$\mathcal{K}^2_{DA}(N) = O_{i.o.}(f(N))$ for $f(N)$ any non-decreasing unbounded computable function.*

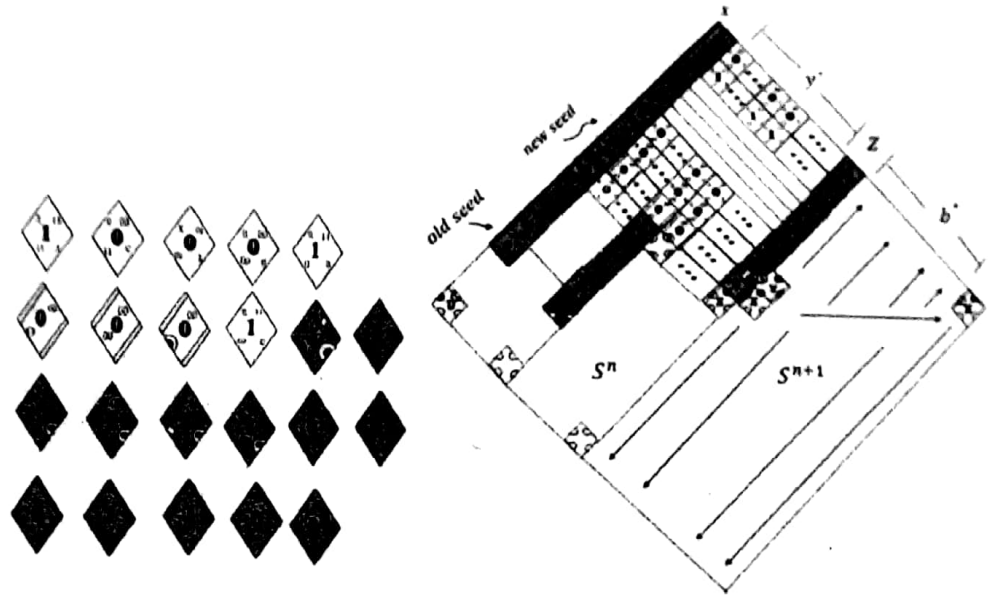


Fig. 6. Formation of an $N \times N$ diamond using $O_{t.o.}(log^* N)$

Proof. *Our proof relies on a self-assembly version of the Busy Beaver problem* [Ra1]. *Define:*

$$B^{\mathcal{T}}_{DA}(n) = max\{N s.t. N, n) \in D^{\mathcal{T}}\}.$$

To prove Theorem 5, we first show

$$B^2_{DA}(n) = \Omega(F(n)) \quad \text{for any computable function} \quad F(n). \tag{1}$$

Theorem 5 follows from (1) by contradiction: if false, then there exists a computable, non-decreasing, unbounded function $f(N)$ such that $\exists\ N_0$ s.t. $\forall\ N > N_0$, $\mathcal{K}^2_{DA}(N) \geq f(N)$.

Let $F(n) = max\ \{N\ s.t\ N = 0\ or\ f(N) \leq n\}$; this is a computational function. Note that $B^2_{DA}(n) \geq F(n)$ requires that $\exists\ (N, n) \in D^2$ s.t. $N \geq F(n)$ and therefore $f(N) > n$ and $\mathcal{K}^2_{DA}(N) \leq n$. For $N > N_0$ this contradicts $\mathcal{K}^2_{DA}(N) \geq f(N)$. Therefore, for all $n > f(N_0)$, $B^2_{DA}(n) < F(n)$, contradicting (1) and establishing Theorem 5.

Recall that $B_t(m) = \Omega(F'(m))$ for any computable function $F'(m)$ where:

$$B_t(m) = max\{t\ s.t.\ m = qs \text{ and there exists a } q\text{-state, } s\text{-symbol Turing machine that halts on a blank tape in } t \text{ steps}\}$$

Let M be a q-state, s-symbol Turing machine that halts on a blank tape in $B_t(m)$ steps, where $m = qs$. We will construct a diamond of size $N = 2B_t(m) + 3$ using $n = 12qs + 4s + 9$ diamond tiles by simulating M with tiles, similar to the construction of Robinson [Ro1]. Given any $n > 41$, we will use $s_n = 2$, $q_n = \lfloor \frac{n-17}{24} \rfloor$ and $m_n = q_n s_n$; our construction will need only $12q_n s_n + 4s_n + 9 < n$ diamond tiles. Then $B^2_{DA}(n) \geq 2B_t(m_n)+3 = \Omega(F'(m_n))$. For any computable

function $F(n)$, we can find another computable function $F'(m)$ s.t. $\forall\, n$, $F'(m_n) > F(n)$. Therefore, we arrive at (1). In Fig. 7, The Busy Beaver machine simulated here has three states ($q_0 = A$, $q_1 = B$, $q_2 = C$) and two symbols ($s_0 = 0, s_1 = 1$). Note that R denotes right, L denotes left and $4x$ indicates that four variations of a tile are used, one for each compass direction.

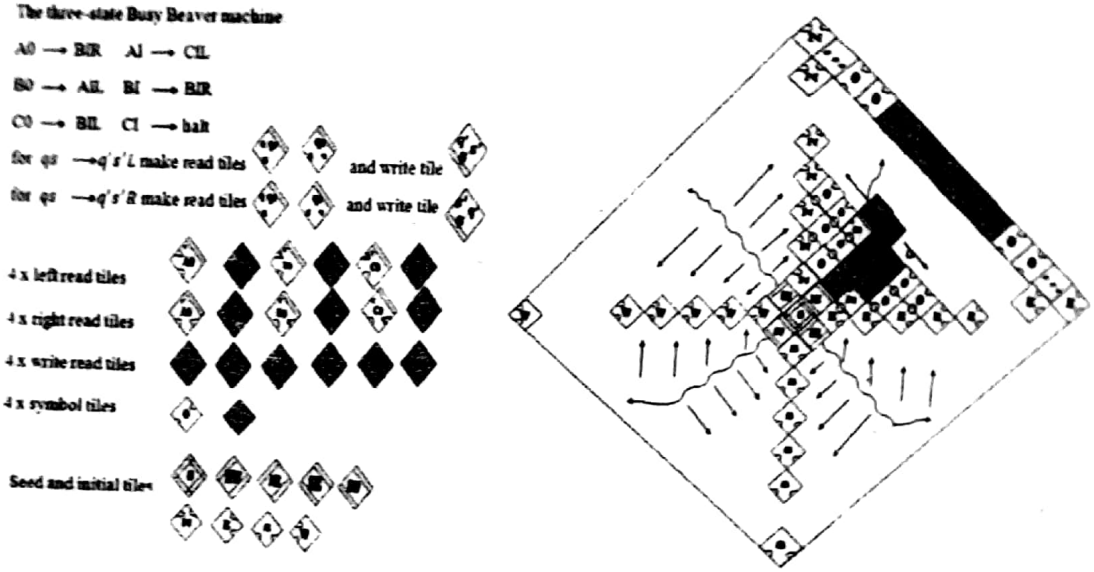


Fig. 7. Formation of an $N \times N$ diamond by growing four identical simulations of a given Turing machine.

We construct the diamond by growing four identical simulations of the Turing machine M, one from each side of a seed tile. Each simulation stays within one of the four regions bounded by the diagonals of the diamond; when M halts, the diamond is complete. We require 4 diamond tiles to create the four half-diagonals defining these boundaries between simulations. For each simulation we require 1 initial state that matches the seed diamond tile, s symbol diamond tiles, qs write diamond tiles and $2qs$ read diamond tiles giving a total of $3qs + s + 1$ diamond tiles per simulation. We describe these diamond tiles for the TM simulation to the northeast of the diamond seed tile. Recall that a diamond tile is a 4-tuple $(\sigma_{NW}, \sigma_{NE}, \sigma_{SE}, \sigma_{SW})$ representing the northwest, northeast, southeast and southwest binding domains. Binding domain strengths are 1 unless noted. Each of the four simulations has its own version of the side labels described, distinguished by superscripts (we omit the superscript ^{N} from the description of northwest facing simulation below).

The symbol diamond tile for symbol s is $(\sigma_s, \sigma_e, \sigma_s, \sigma_e)$ where σ_s is a binding domain representing the symbol s and σ_e is a binding domain indicating that the TM head is not present. For each state-symbol pair (q, s) the left read diamond tile $(\sigma_{q,s}, \sigma_e, \sigma_s, \sigma_q)$ and the right read tile $(\sigma_{q,s}, \sigma_q, \sigma_s, \sigma_e)$ represent the TM head in state q entering a tape cell (from the left or from the right) and reading the symbol s. The binding domain $\sigma_{q,s}$ have strength 2; this is necessary for the TM head to enter the next row of the simulation. The write diamond tiles, representing the action the TM head takes depend on the form of the state transition table entry. For each entry of the form $(q, s) \to (q', s', L)$ there is a write

diamond tile $(\sigma_{s'}, \sigma_e, \sigma_{q,s}, \sigma_{q'})$*; for each entry of the form* $(q, s) \to (q', s', R)$ *there is a write diamond tile* $(\sigma_{s'}, \sigma_{q'}, \sigma_{q,s}, \sigma_e)$*; for each entry of the form* $(q, s) \to halt$ *there is a write diamond tile* $(\sigma_{halt}, \sigma_e, \sigma_{q,s}, \sigma_e)$.

To start the Turing Machine in state q_0 *reading the blank symbol* s_0*, the initial tile for the northeast simulation is* $NE = (\sigma_{q_0,s_0}, \sigma_e, \sigma_S, \sigma_e)$ *where* σ_S *is a strength-2 binding domain. The initial diamond tiles for all four simulations bind to the diamond seed tile* $S = (\sigma_S^{NW}, \sigma_S^{NE}, \sigma_S^{SE}, \sigma_S^{SW})$*. The four diagonal diamond tiles,* $N = (\sigma_{s_0}^N, \sigma_e^N, \sigma_e^W, \sigma_{s_0}^W)$*,* $E = (\sigma_{s_0}^N, \sigma_{s_0}^E, \sigma_e^E, \sigma_e^N)$*,* $S = (\sigma_e^E, \sigma_{s_0}^E, \sigma_{s_0}^S, \sigma_e^S)$ *and* $W = (\sigma_e^W, \sigma_e^S, \sigma_{s_0}^S, \sigma_{s_0}^W)$ *pad the tapes with extra cells containing the blank symbol* s_0 *and delimit the four simulations.*

4 Conclusion

This paper discussed the program-size complexity of self-assembled diamonds, where complexity is measured by the number of distinct diamond tile types involved. An alternative complexity measure is the minimum number of distinct side labels required uniquely to produce the object. The number of labels will be relevant in a physical system where the number of distinct binding interactions are limited due to imperfect specificity of binding. A main conclusion of this paper is that the program-size complexity of self-assembled objects (at $\mathcal{T} = 2$) looks remarkably similar to the usual program-size complexity with respect to Turing Machines.

References

[LL1] Lathrop, J.I., Lutz, J.H., Patitz, M.J., Summers, S.M.: Computability and complexity in self-assembly. Theory Comput. Syst. **48**(3), 617–647 (2011)

[Ra1] Rado, T.: On non-computable functions. Bell Syst. Tech. J. **41**(3), 877–884 (1962)

[RE1] Rothemund, P.W.K., Winfree, E.: The program-size complexity of self-assembled squares (extended abstract), Oregon, USA, pp. 1–37 (2000)

[Ro1] Robinson, R.M.: Undecidability and non periodicity of tilings of the plane. Inventions Math. **12**, 177–209 (1971)

[Se1] Seeman, N.C.: DNA nanotechnology: novel DNA constructions. Annu. Rev. Biophys. Biomol. Struct. **27**, 225–248 (1998)

[SE1] Soloveichik, D., Winfree, E.: Complexity of self-assembled shapes. SIAM J. Comput. **36**(6), 1544–1569 (2007)

[Su1] Summers, S.M.: Reducing tile complexity for the self-assembly of scaled shapes through temperature programming. Algorithmica **63**(1–2), 117–136 (2012)

[Wa1] Wang, H.: Proving theorems by pattern recognition, II. Bell Syst. Tech. J. **40**, 1–42 (1961)

[Wi1] Winfree, E.: Simulations of computing by self-assembly. In: Proceedings of the Fourth DIMACS Meeting on DNA Based Computers, pp. 1–27 (1998)

[Wi2] Winfree, E.: Algorithmic self-assembly of DNA. Ph.D. thesis, California Institute of Technology, June 1998

[WL1] Winfree, E., Liu, F., Wenzler, L.A., Seeman, N.C.: Design and self-assembly of two-dimensional DNA Crystals. Nature **394**, 539–544 (1998)