Homework 1 – ECS 289, Theory of Molecular Computation
Winter 2021

Please submit a PDF file to Canvas. You may work in groups of two if you like, or you may work on your own. The purpose of group work is that, as in research, sometimes it helps to talk through a problem with another person to help you think about how to solve it. The purpose is emphatically **not** to split up the problems so that you don't have to think about some of the problems at all; this is not permitted. Even if one person has the key idea, or takes the lead on writing up the solution, both partners should participate in creating and writing up the solution.

Do not put your name(s) on the homework, because we will do anonymous peer review. The peer reviews will be due one week after the assignment is due, and all peer reviews are done individually.

For the problems where you are creating a tile assembly system, use the ISU TAS simulator (`http://self-assembly.net/wiki/index.php?title=ISU_TAS`), or another simulator if you like (e.g., VersaTile or xgrow). Please include

- a text description in the PDF file describing how the tile system works,
- screenshots in the PDF file from the simulator showing examples of producible and terminal assemblies (i.e., make it easy to grade and verify it works in simulation), and
- the TAS input files (uploaded separately to Canvas) so that your tile assembly system can be tested in the simulator if necessary.

To aid with peer review, there are actually separate Canvas assignments for each problem. Please upload each solution individually. This way, when we assign peer review, you won't be reviewing the same person/group for all the assigned problems.

Required problems

spell Γ: Design a tile set to "spell Γ," with size programmed by the seed. More precisely, for each $n \geq 2$, let σ_n be the seed assembly shown in Figure 1. Design a tile set T such that for all $n \geq 2$, the tile assembly system $\mathcal{T} = (T, \sigma_n, 2)$ produces the shape Γ_n shown by example in Figure 2.

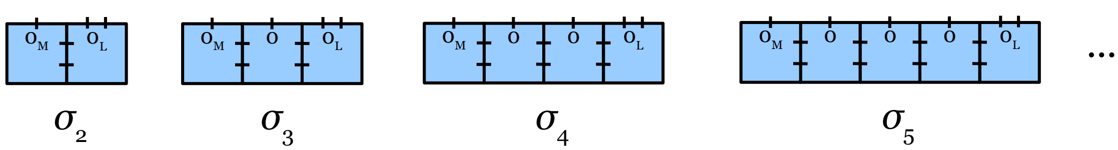

Figure 1: Double-notches represent strength-2 glues and single-notches represent strength-1 glues.

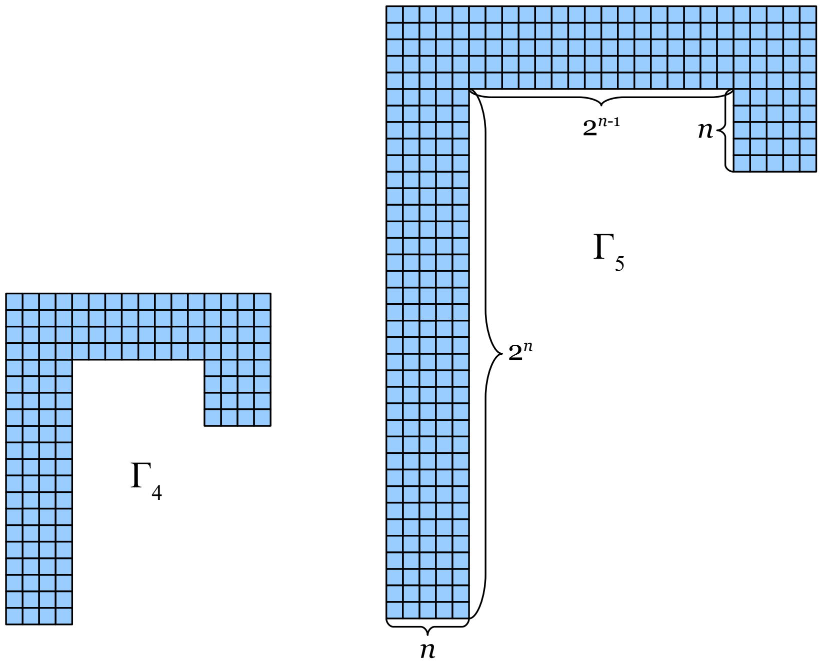

Figure 2: Γ_n shapes for $n = 4, 5$.

busy tiles: We saw there is a way to make an $n \times n$ square from $O(\log n / \log\log n)$ tile types, and that some n require at least that many tile types. In particular, the size of the final structure (n^2 tiles) is exponentially larger than the number of tile types. This is an open-ended exercise: devise a singly-seeded tile system that creates an even larger finite assembly α relative to the number of tile types $|T|$ it requires. (Not necessarily a square.) The bigger the ratio $\frac{|\alpha|}{|T|}$, the better. (If you want a concrete goal to shoot for, try for ratio $\approx 2^{2^{|T|}}/|T|$.)

computability: Show that every set $X \subseteq \mathbb{Z}^2$ weakly self-assembled by a tile system is computably enumerable. Discuss why your proof fails to show that X is computable. (**hint:** if your proof actually works to show computability, i.e., if you give an algorithm for X that always halts, then something's wrong) Write down explicit pseudocode for your algorithm.

There are many equivalent definitions of computably enumerable, and you can use any of them. (e.g., one is that there is an algorithm that, on input $p \in \mathbb{Z}^2$, halts if $p \in X$ and does not halt if $p \notin X$.)

transitivity of producibility: Show that for any tile assembly system $\mathcal{T}$, the relation $\to^{\mathcal{T}}$ is transitive. In other words, for assemblies α, β, and γ, if $\alpha \to \beta$ and $\beta \to \gamma$, then $\alpha \to \gamma$. This is quite straightforward if $S_\beta \setminus S_\alpha$ is finite, i.e., a finite assembly sequence can grow α into β. (Even if S_α is infinite, this case is easy since only finitely many tiles are added.) The hard part is to show it holds even if β is infinitely larger than α. However, as a first step, it helps to write down a detailed proof for the finite case.[1]

[1] *Detailed* here means don't just write "*first produce β from α, then produce γ from β*", even though that's intuitively how the simple proof works. Instead, go down to the definitions: based on the definition of $\to$, you need

Optional problems

You don't have to do these, but I think they are interesting to think about. Feel free to discuss them on Campuswire with other students.

tile complexity of trees: Suppose we have a finite tree shape $\Upsilon \subset \mathbb{Z}^2$ (i.e., one with no cycles). What is the tile complexity of Υ?

Hint: Consider fixing the position p where the seed goes, and think of Υ as a *rooted* tree, with root p. Characterize the tile complexity $C_p(\Upsilon)$, subject to the seed being at p, by structural induction on this rooted tree. For example, the base case is a leaf; how many tile types are needed for all the leaves of Υ? Then the tile complexity of Υ is $\min_{p \in S_\Upsilon} C_p(\Upsilon)$.

directed: Show that the following definitions of "$\mathcal{T}$ *is directed*" are all equivalent. (This was claimed but not proven in the lecture slides.)

1. $|\mathcal{A}_\Box[\mathcal{T}]| = 1$.
2. For all $\alpha, \beta \in \mathcal{A}[\mathcal{T}]$, there exists $\gamma \in \mathcal{A}[\mathcal{T}]$ such that $\alpha \to \gamma$ and $\beta \to \gamma$.
3. For all $\alpha, \beta \in \mathcal{A}[\mathcal{T}]$ and $\vec{p} \in S_\alpha \cap S_\beta$, $\alpha(\vec{p}) = \beta(\vec{p})$.

diamonds are forever: Let $\mathcal{D} = \{D_0, D_1, D_2, D_3, \ldots\}$ be the infinite set of shapes shown in Figure 3, where for each $n \in \mathbb{N}$, $D_n = \left\{ \ (x, y) \in \mathbb{Z}^2 \ \middle| \ |x| + |y| \leq n \ \right\}$ is the set of points with Manhattan distance (a.k.a., L_1 distance) at most n from the origin (i.e., the "circle" of radius n in the L_1 metric, which looks like a diamond).

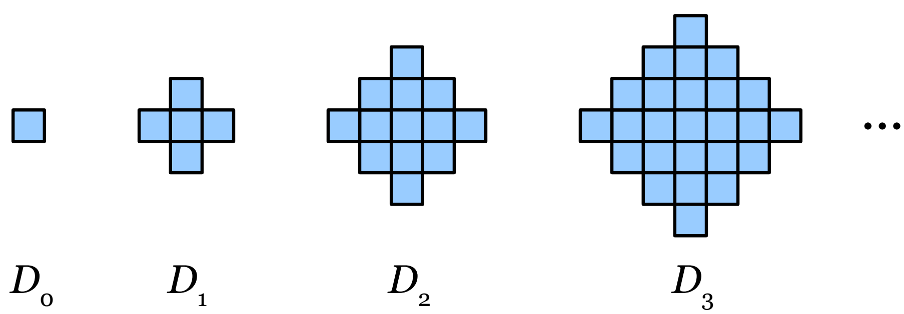

Figure 3: Infinite family of diamond shapes.

Say a tile assembly system $\mathcal{T} = (T, \sigma, \tau)$ is *singly-seeded* if $|\text{dom}\ \sigma| = 1$. Design a singly-seeded tile assembly system $\mathcal{T}$ such that, defining

$$A = \{ \ S_\alpha \ \mid \alpha \in \mathcal{A}_\Box[\mathcal{T}] \text{ and } |S_\alpha| < \infty \ \},$$

to show that given an assembly sequence s_1 starting at α, with result β, and another assembly sequence s_2 starting at β, with result γ, you can create an assembly sequence s_3 starting at α, with result γ. How exactly do you define s_3 based on s_1 and s_2 and argue that it is a valid assembly sequence? If the simple proof for s_1 being finite is sufficiently detailed, you should be able to point to the part that would break if s_1 were infinite, which will help you to see what needs to be handled in the case where s_1 is infinite.

we have $|A| = \infty$ and $A \subseteq \mathcal{D}$. In other words, the set of shapes of finite terminal assemblies of $\mathcal{T}$ is an infinite subset of $\mathcal{D}$.

Make A the largest subset of $\mathcal{D}$ that you can. Explain the difficulty with trying to make it larger.

arbitrarily large assemblies $\implies$ infinite assembly: Show that for any tile assembly system $\mathcal{T}$ satisfying the problem **diamonds are forever**, there is an infinite assembly $\alpha \in \mathcal{A}_\square[\mathcal{T}]$. (Note that each $D_i \in \mathcal{A}_\square[\mathcal{T}]$ is finite.)

cellular automata: Design a tile assembly system to simulate cellular automaton Rule 110:

`https://en.wikipedia.org/wiki/Rule_110#Definition`

Defining what exactly it means to "simulate" a cellular automaton is part of the problem. Explain the sense in which your tile system "simulates" Rule 110, and in particular, if there are any ways it could be improved that you couldn't figure out how to do. (Think of this part as practice for the art of stating open questions at the end of a research paper.)

hailstone: Design a tile set so that, at temperature 2, if initialized with a horizontal, 1D seed assembly of length $\approx \log n$ representing a positive integer n in binary in its glues, computes successive applications of the *Collatz function* (a.k.a., the *hailstone function*) $f : \mathbb{N} \to \mathbb{N}$ defined for all $n \in \mathbb{N}$ by:

$$f(n) \begin{cases} 3n+1, & \text{if } n \text{ is odd;} \\ n/2, & \text{if } n \text{ is even.} \end{cases}$$

What this means is: if the seed row represents n, then after some amount of growth, eventually a row appears representing $f(n)$, then after some more growth, a row appears representing $f(f(n))$, and so on.

The tile system should *stop* growing if ever it reaches the value 1.

guaranteed nondeterminism: Show that there is a finite shape $S \subset \mathbb{Z}^2$ so that every singly-seeded tile assembly system that strictly self-assembles S has more than one terminal assembly sequence. In other words, there will always be nondeterministic *order of binding*. Note that this is *different* from having more than one terminal assembly: a TAS can have multiple assembly sequences that all result in the same terminal assembly.

guaranteed determinism: Characterize exactly the set of finite shapes that can be assembled by a singly-seeded tile assembly system with exactly one terminal assembly sequence.

binding with no excessive strength: Show that every finite shape can be strictly self-assembled by a singly-seeded tile assembly system $\mathcal{T} = (T, \sigma, \tau)$ such that every tile binds with strength exactly τ (in other words, there are never any binding events with *excessive* binding strength). Note that this refers only to initial binding; the tile may *later* have tiles attach adjacent to it so that its total bond strength with the rest of the assemble exceeds τ.

binding with excessive strength: Show that there is a finite shape $S \subset \mathbb{Z}^2$ such that the *smallest* singly-seeded tile assembly system $\mathcal{T} = (T, \sigma, \tau)$ strictly self-assembling S permits tiles to bind with strength strictly larger than τ. (In other words, although it is possible to assemble S while guaranteeing no excessive strength binding, one can use fewer tile types if excessive strength binding is allowed.)

finding the end: Let $\sigma_1, \sigma_2, \sigma_3, \ldots$ be a infinite sequence of assemblies of the form shown in Figure 4, where in each case, assume the leftmost tile is at the origin.

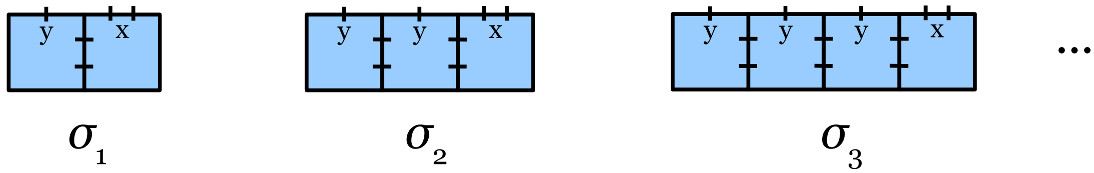

Figure 4: Infinite family of "floor" seeds. Double-notches represent strength-2 glues and single-notches represent strength-1 glues.

Call the three tile types used t_1, t_2, t_3 . For example, σ_3 has t_1 on the left, t_3 on the right, and two copies of t_2 in the middle. For any tile set T and any $n \in \mathbb{Z}^+$, define the tile system $\mathcal{T}_{T,n} = (T \cup \{t_1, t_2, t_3\}, \sigma_n, 2)$.

Imagine the following goal: to design a single tile set T such that, for all $n \in \mathbb{Z}^+$, $\mathcal{T}_{T,n}$ is guaranteed to place a tile at position $(-1, 0)$ (i.e., just to the left of the leftmost seed tile), without ever placing a tile underneath the seed. T is trying to figure out where the floor ends and place a tile there.

Show that for any such tile set T, there exists $n \in \mathbb{Z}^+$ such that $\mathcal{A}[\mathcal{T}_{T,n}]$ has an infinite assembly. (In other words, there is no way to "search" for where the floor ends, "sense" it, and stop growing; the only way to guarantee a tile is placed at $(-1, 0)$ is to place infinitely many tiles.)

Note that this task *is* possible if we restrict the problem in one of two ways. If we require the tile system only to work for *even-length* seeds, then it can be done. Or, if we allow the tile system to *grow underneath the seed* in addition to above the seed, then it can also be done.