Synthesizing Small and Reliable Tile Sets for Patterned DNA Self-Assembly

Tuomo Lempiäinen, Eugen Czeizler, Pekka Orponen

Department of Information and Computer Science
Aalto University, Finland
email: {tuomo.lempiainen, eugen.czeizler, pekka.orponen}@aalto.fi

Aalto University
School of Science

Algorithmic self-assembly, a ground-breaking technology

Self-assembly of nanostructures templated on synthetic DNA has been proposed by several authors as a potentially ground-breaking technology for the manufacture of next-generation circuits, devices, and materials.

The abstract Tile Assembly Model (aTAM) is a custom designed formalization for the study of self-assembly systems. Tiles can be thought of as non-rotating unit squares with different kinds of glues associated with their edges. Tile assemblies of patterns are achieved through the process of self-assembly.

A given target pattern can be assembled from many different families of base tiles. Thus, it is advantageous to:

- Minimize the number of tile types needed;
- Maximize the probability that they self-assemble into the desired pattern.

The Patterned self-Assembly Tile set Synthesis (PATS) Problem

Given a rectangular 2D pattern of colored tiles, find a minimal set of tile types which self-assemble into that pattern (in the aTAM framework).

- Proposed by Ma and Lombardi (2008) (also showed it is NP-hard in 2009);
- Göös and Orponen (2010) presented an exhaustive partition search branch-and-bound algorithm (PS-BB) for it.

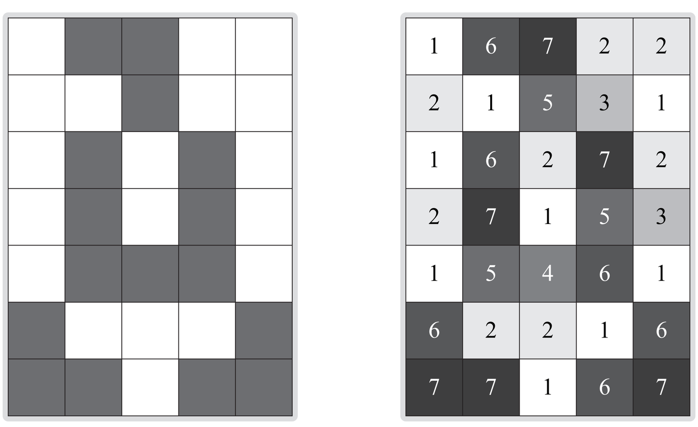
If we are interested in finding small but not provably minimal solutions, we can do better than the PS-BB algorithm.

- By applying a new heuristic in traversing the search space, we often have a considerable improvement in the size of the solutions;
- By running several instances of the algorithm in paraller, we increase our probability of success.

The PS-BB algorithm

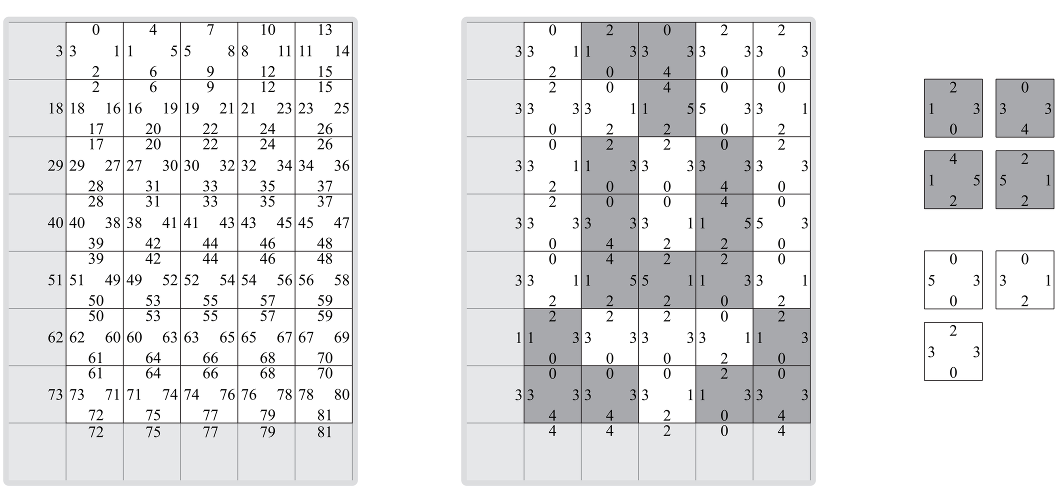
The algorithm is based on the following key ideas:

- Perform an exhaustive search in the lattice of partitions of the rectangle $[m] \times [n]$;
- For a partition P, test whether it can be generated by some deterministic tile set;
- If so, then proceed to consider coarsenings of P, i.e. smaller tile sets, by merging two partition classes. If not, then backtrack;
- Use several effective bounding methods to prune the branches of the search.



(a) (b)

Fig. 1. (a) Partition A. (b) A partition M that is a refinement of A with |M| = 7 parts.



(a) (b) (c)

Fig. 2. (a) A most general tile assignment for the initial partition (with a seed assembly in place). (b) Finished assembly for the pattern from Fig. 1a. The tile set to construct this assembly is given in (c).

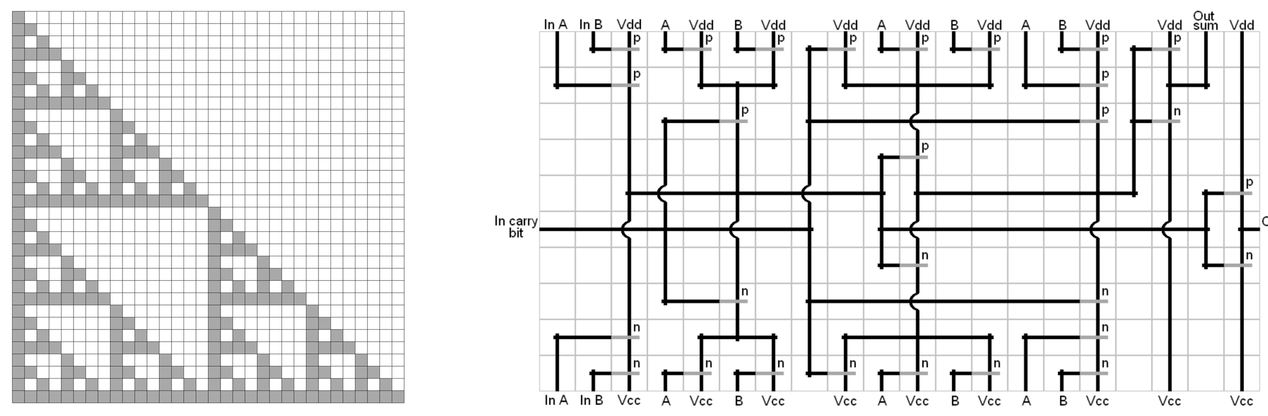
The PS-H algorithm

Instead of trying to reduce the size of the search space, our new PS-H algorithm attempts to "greedily" optimize the order in which the coarsenings of a partition are explored, in the hope of being directly lead to close-to-optimal solutions.

The basic heuristic idea is to try to minimize the effect that a merge operation has on other partition classes than those which are combined. We choose the pair of classes to be merged using the following criteria:

- Primarily, prefer pairs having as many common glues as possible;
- Secondarily, prefer larger classes over smaller ones;
- In case of several equally good pairs, pick one at random.

Because of the randomization, some of the runs lead to small solutions quickly, while others get sidetracked into worthless expanses of the solution space. We utilize this by running n instances of the algorithm in parallel and denote that with PS-H$_n$.



(a) (b)

Fig. 3. (a) A 32×32 subset of the Sierpinski pattern. (b) A CMOS full adder that induces a 15-color 20×10 pattern.

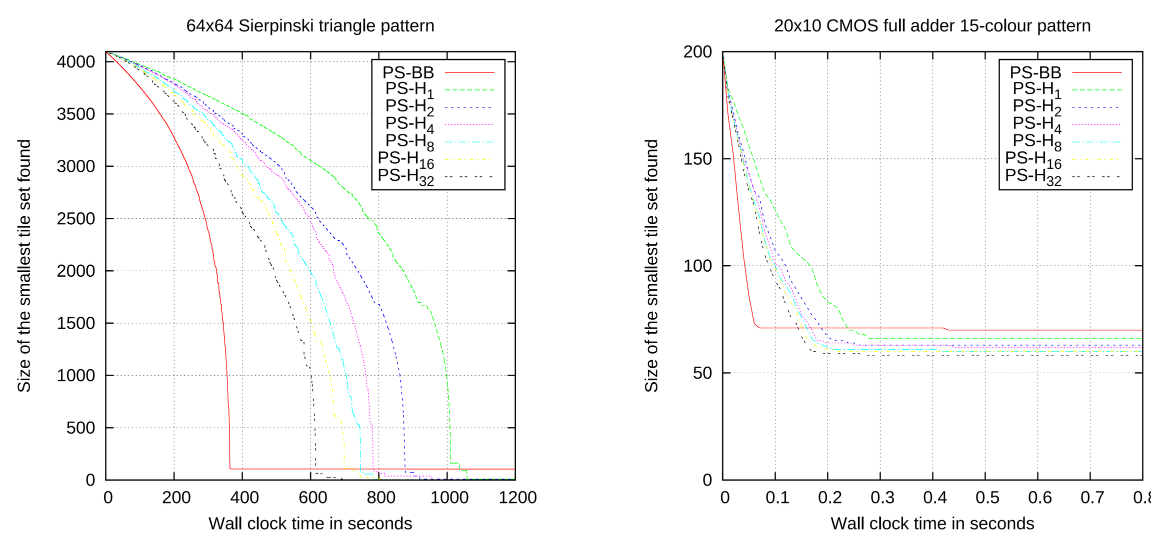


Fig. 4. Evolution of the smallest tile set found by the PS-BB and PS-H algorithms as a function of time.

Synthesizing reliable tile sets

Given the amount of time the assembly process is allowed to take, we define the *reliability of a tile set* to be the probability that the assembly process of the tile set in question completes without any incorrect tiles being frozen in the terminal configuration.

Based on Winfree's analysis of the kinetic Tile Assembly Model (1998), we can assess the reliability of different tile sets. The assembly process can be thought of as a sequence of tile addition steps $(a_1, a_2, \ldots, a_N)$, where $a_k = (i_k, j_k)$ denotes a tile being frozen on site (i_k, j_k). Then, the probability of perfect assembly is

$$\begin{aligned}\Pr(correct\ pattern) &= \Pr(C(a_1) \cap C(a_2) \cap \cdots \cap C(a_N)) \\ &= \prod_{i,j} \Pr(C(i,j) \mid C(i-1,j) \cap C(i,j-1)),\end{aligned}$$

where

$$\Pr(C(i,j) \mid C(i-1,j) \cap C(i,j-1)) = \frac{\frac{1}{r^*+r_{r,2}}}{\frac{1}{r^*+r_{r,2}} + \frac{M^1_{ij}}{r^*+r_{r,1}} + \frac{M^2_{ij}}{r^*+r_{r,0}}}.$$

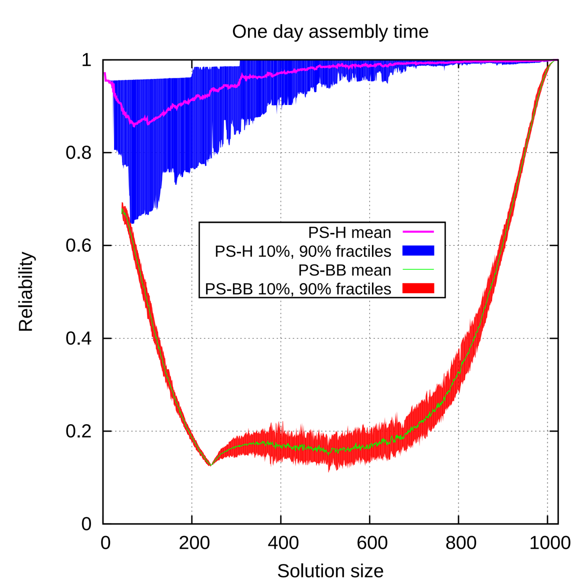


Fig. 5. Reliability of the solutions for the 32×32 Sierpinski pattern found by the PS-H and PS-BB algorithms, allowing assembly time of one day. Note that the smallest tile sets are also the most reliable ones.

Statistical Mechanics and Computation of DNA Self-Assembly, Mariehamn 2011