An introduction to self-assembly
Theory and experiments

Pierre-Étienne Meunier

Postdoc, LIF, Aix-Marseille Université

Theory matters, so do experiments

Theory is concerned with *theorems*, i.e. facts that are derived from axioms by a logical process. Theorems are *certainly* true, but they work on abstract systems.
They can be used:

- to answer to *"what"*, *"why"*, and *"is it possible"* sorts of questions.
- to find the boundaries of a system.

Experiments are tests against natural, unknown phenomena. They are used to observe things, i.e. to answer to *"what"* kinds of questions.

Why does it matter at all?

In computer science, things are mixed up: computers are real things, yet they come from an answer to an *"is it possible"* question.

Algorithms everywhere!

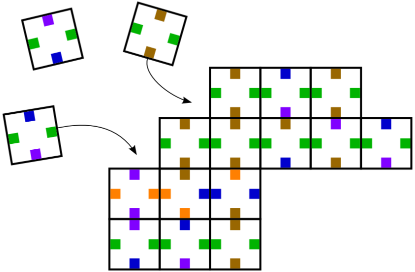
An algorithm is a dynamic process, related to the flow of information (mutations, signaling pathways, geometric constraints…) in a system.

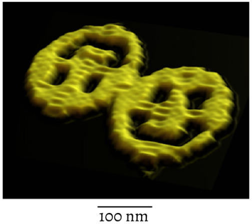
Does that make us computer scientists: experimentalists, engineers, or theorists?

My biggest differences with biologists right now:

1. data
2. bottom-up versus top-down

Tiling





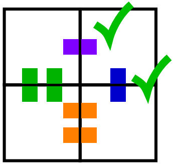
Atomic Force Microscope picture
Paul Rothemund 2006

At the stage of experimentation : fractal structures, nano-robots...

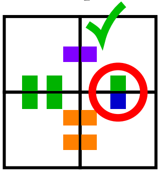
Assembly rules

Important parameter: the temperature

Temperature 2
(Cooperative)



Temperature 1
(Non-cooperative)



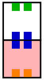
At least the temperature

$\tau = 2.$

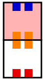
$\tau = 1.$

At least the temperature

$\tau = 2.$

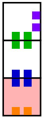


$\tau = 1.$

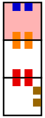


At least the temperature

$\tau = 2.$

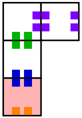


$\tau = 1.$

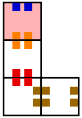


At least the temperature

$\tau = 2.$

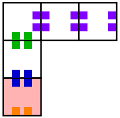


$\tau = 1.$

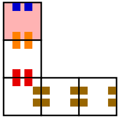


At least the temperature

$\tau = 2.$

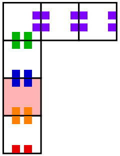


$\tau = 1.$

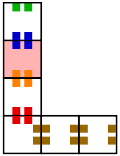


At least the temperature

$\tau = 2.$

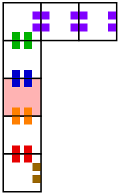


$\tau = 1.$

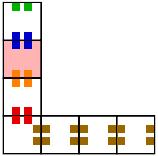


At least the temperature

$\tau = 2.$

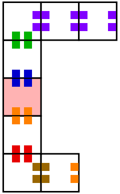


$\tau = 1.$

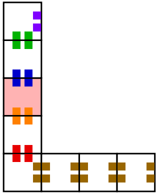


At least the temperature

$\tau = 2$.

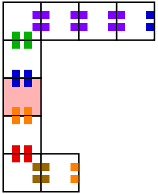


$\tau = 1$.

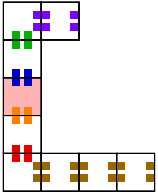


At least the temperature

$\tau = 2$.

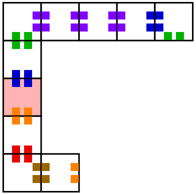


$\tau = 1$.

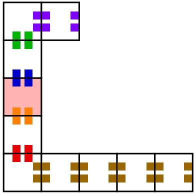


At least the temperature

$\tau = 2$.

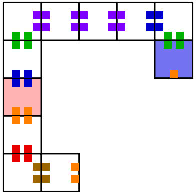


$\tau = 1$.

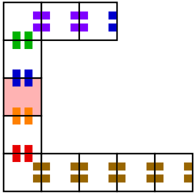


At least the temperature

$\tau = 2$.

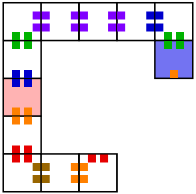


$\tau = 1$.

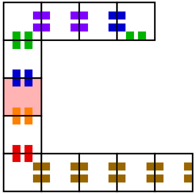


At least the temperature

$\tau = 2$.

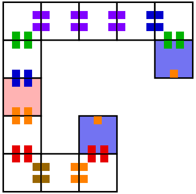


$\tau = 1$.

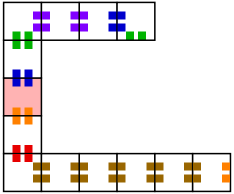


At least the temperature

$\tau = 2.$

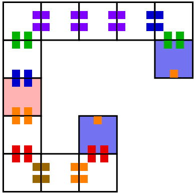


$\tau = 1.$

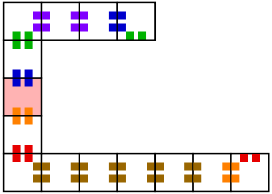


At least the temperature

$\tau = 2$.

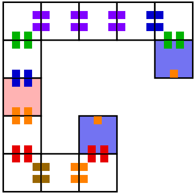


$\tau = 1$.

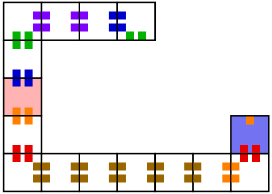


At least the temperature

$\tau = 2$.

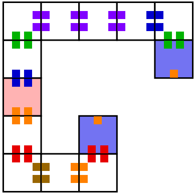


$\tau = 1$.

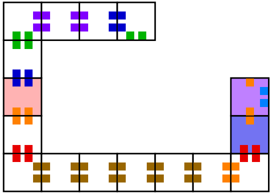


At least the temperature

$\tau = 2.$

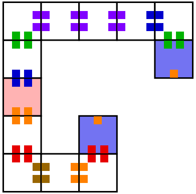


$\tau = 1.$

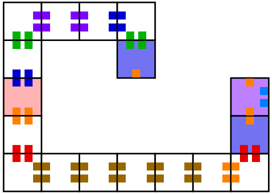


At least the temperature

$\tau = 2.$

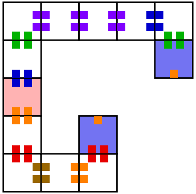


$\tau = 1.$

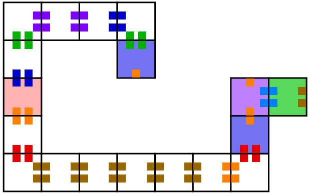


At least the temperature

$\tau = 2$.

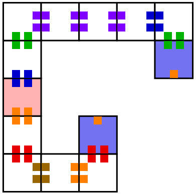


$\tau = 1$.

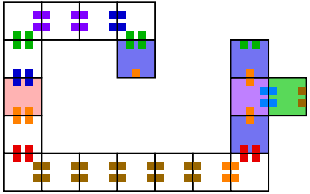


At least the temperature

$\tau = 2$.

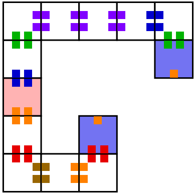


$\tau = 1$.

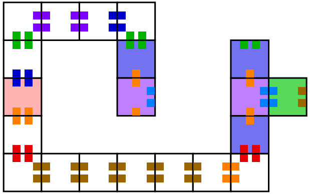


At least the temperature

$\tau = 2$.

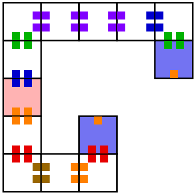


$\tau = 1$.

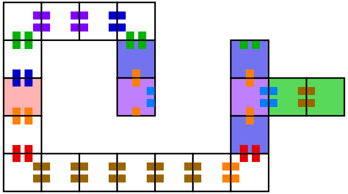


At least the temperature

$\tau = 2$.

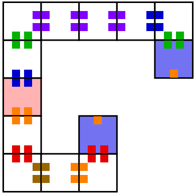


$\tau = 1$.

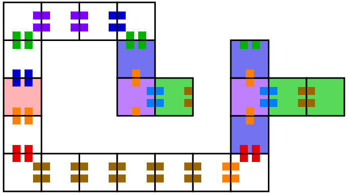


At least the temperature

$\tau = 2$.

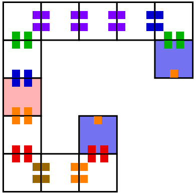


$\tau = 1$.

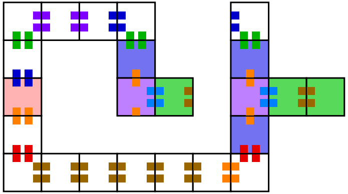


At least the temperature

$\tau = 2$.

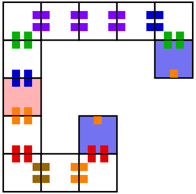


$\tau = 1$.

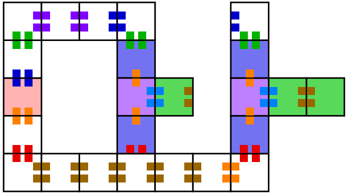


At least the temperature

$\tau = 2$.

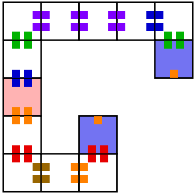


$\tau = 1$.

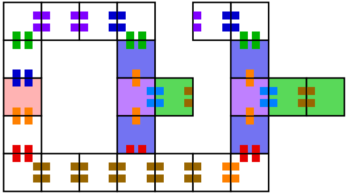


At least the temperature

$\tau = 2$.

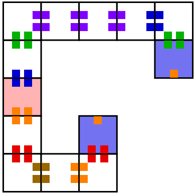


$\tau = 1$.

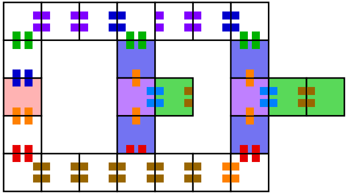


At least the temperature

$\tau = 2$.

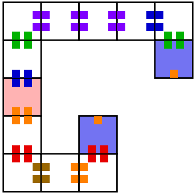


$\tau = 1$.

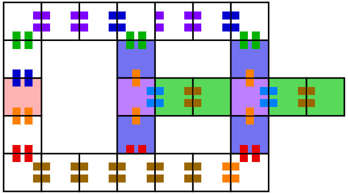


At least the temperature

$\tau = 2.$



$\tau = 1.$



Important results: building things

Theorem 1. (Rothemund, Winfree, STOC 2000)

Building a square of size $n \times n$ requires $O\left(\frac{\log n}{\log \log n}\right)$ tile types.

Theorem 2. (Soloveichik, Winfree, 2007)

Building a shape of Kolmogorov complexity n requires $O(n \log n)$ tile types.

Important results: the hierarchical model

In this model, there are seeds.

But in a solution, tiles do not care whether they are connected to a seed!
Is this a problem?

Theorem 3. (Cannon, Demaine, Demaine, Eisenstat, Patitz, Schweller, Summers, Winslow, 2012)

The two-handed model can simulate the seeded model.

Intrinsic universality

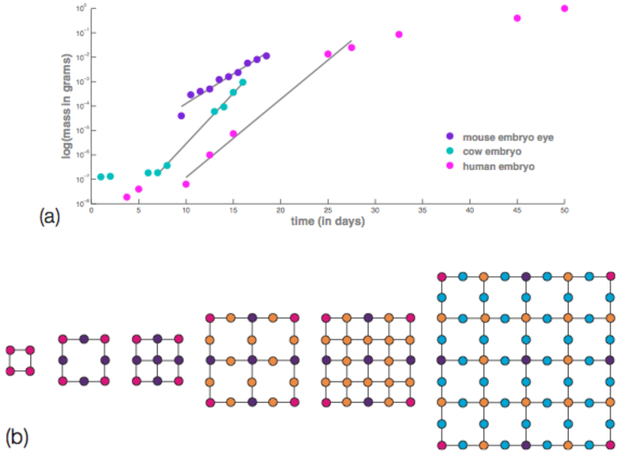
Theorem 4. (Doty, Lutz, Patitz, Schweller, Summers, Woods, FOCS 2012)

There is a single tileset U, that can simulate any other tileset up to rescaling.

Theorem 5. (Demaine (++), Feteke, Patitz, Schweller, Winslow, Woods 2014)

With (complicated) polygons instead of squares, U can be a single tile.

Active self-assembly: nubots



A model with tons of variants

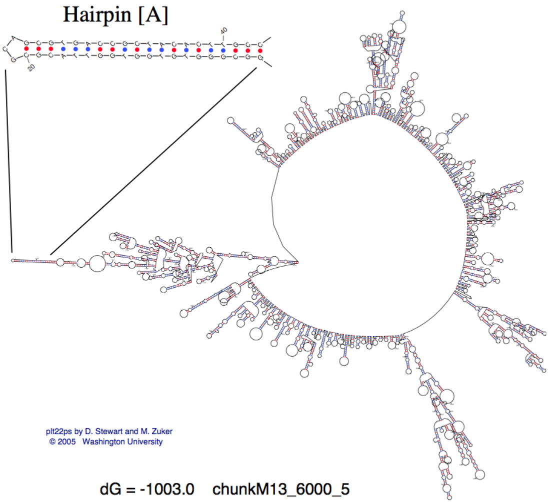
- Staged self-assembly
- Temperature programming
- Error correction with fuzzy temperature
- Concentration programming

Next challenges

- Understanding temperature 1 (almost there)
- Exploring more realistic geometries
- What can be computed with nubots?
- Functional self-assembly: no theoretical model yet. Reproduction modes.
- Experimental challenge: error correction in the tile and nanotubes implementations

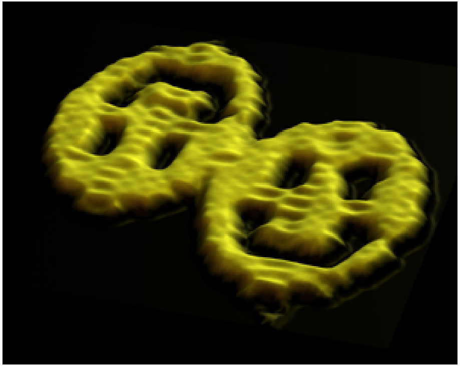
Folding: an ubiquitous computational paradigm

DNA origami (Rothemund, Nature 2006):



Folding: an ubiquitous computational paradigm

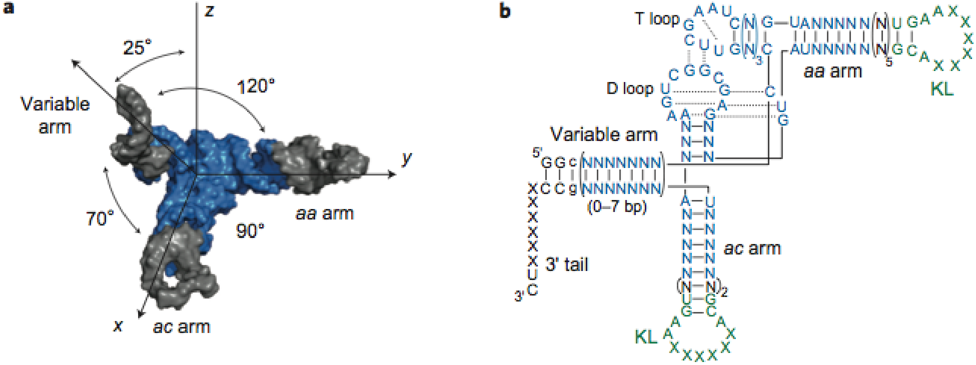
DNA origami (Rothemund, Nature 2006):



Beyond scaffolds and staples

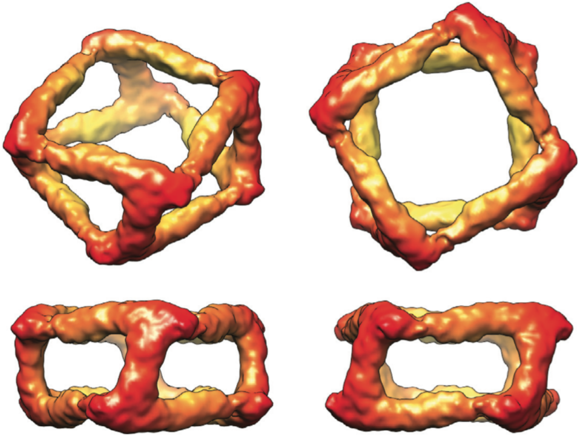
We need techniques for in vivo production: Protein/RNA folding

(Severcan et al, Nature chemistry 2010)



Beyond scaffolds and staples

We need techniques for in vivo production: Protein/RNA folding (Severcan et al, Nature chemistry 2010)

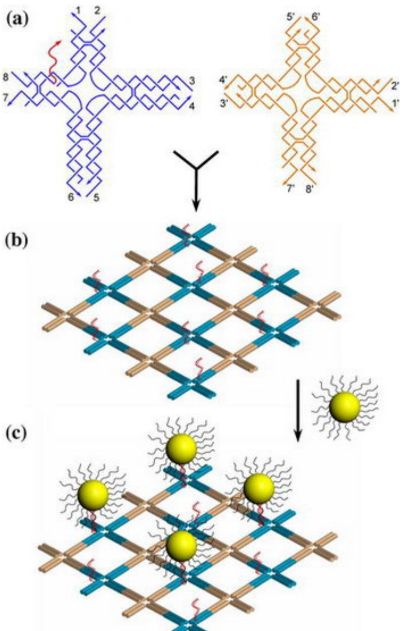


Next challenges

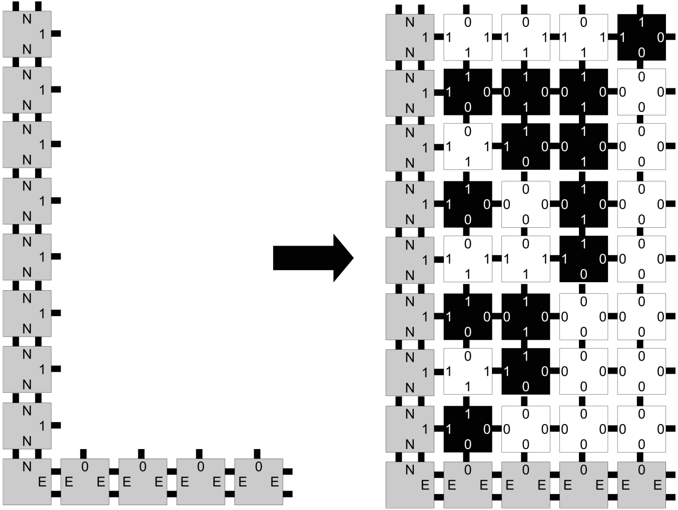
- Using RNA, we can interact with proteins.

- And although we have no theory, it is easier to fold.

Particle placement



Particle placement

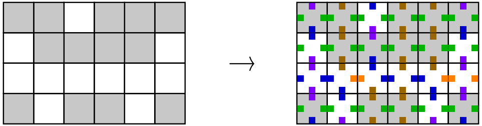


Particle placement

Minimizing the number of tile types:
keeping the scale small, minimizing the error probability.

Pattern Assembly Tileset Synthesis

Find the smallest tileset that produces a given pattern:

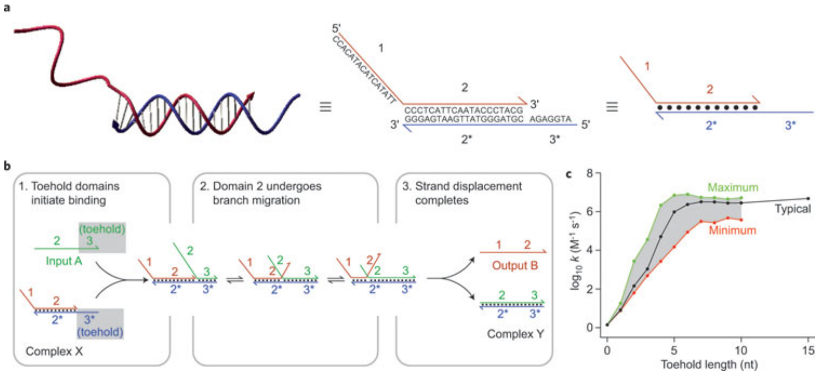


Theorem 6. (Kopecki, Patitz, *Meunier*, Seki, 2014)

NP-complete with only two colors.

By the way, this is (to date) the largest mathematical proof *ever* ($7 \cdot 10^{13}$ cases).

Strand displacement systems



What for?

Theorem 7. (Cook, Soloveichik, Bruck, Winfree, 2009)

Using strand displacement systems, we can simulate:

- arbitrary chemical reaction networks
- arbitrary logic circuits

Challenges:

- Characterizing the possible behaviors of CRN.
- Mastering the implementation (mostly leak and speed).
- Finding a model.
- Combining networks.

Thanks for your attention