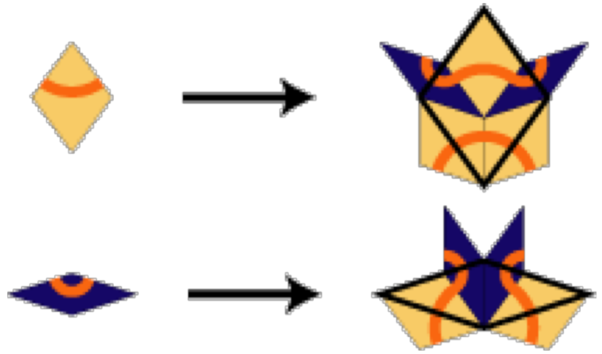
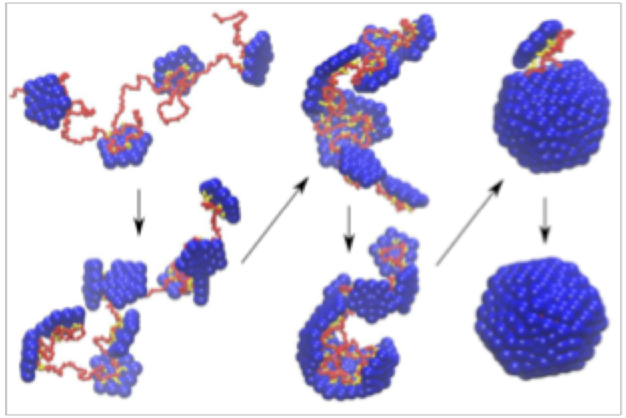
A Survey of Computational Models of Self-Assembly



COT6410 – Spring 2015

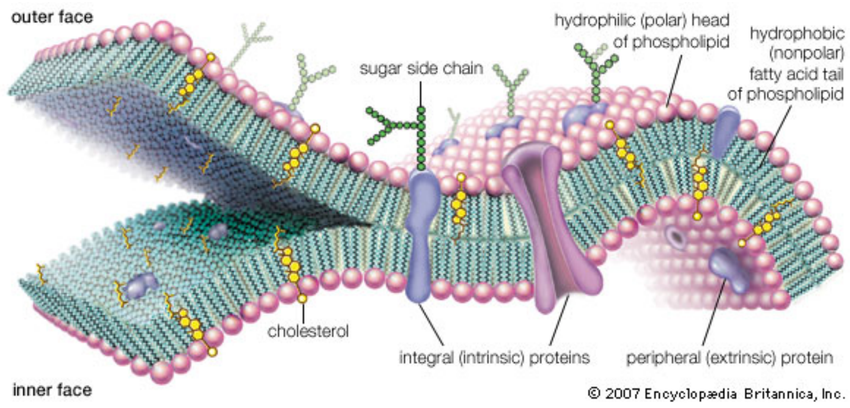


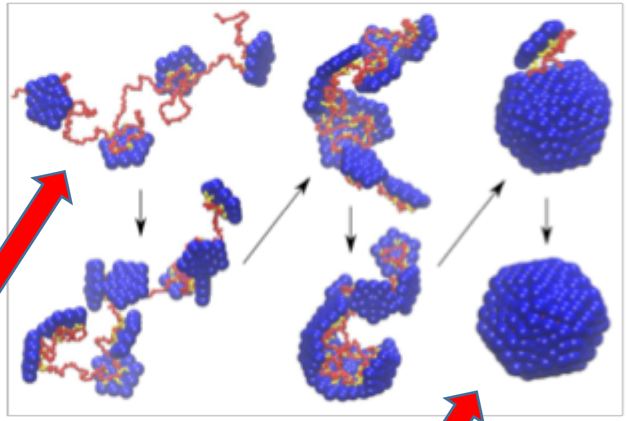
Alternate Presentation Title:

Proof that Theoretical Computer Science isn’t Dead

What is Self Assembly?

- The automatic combination of less complex components into more complex components.
- Sounds crazy? This already happens in nature:
  - Lipids come together to form cell membranes
  - Virus proteins come together to form the capsid, which allows a virus to make you sick.
  - Even crystals “self assemble.”





“Less Complex”

“More Complex”

Algorithmic Construction

"The 'computable' numbers may be described briefly as the real numbers whose expressions as a decimal are calculable by finite means."

-- Alan Turing, *On Computable Numbers*

Am I Computable?

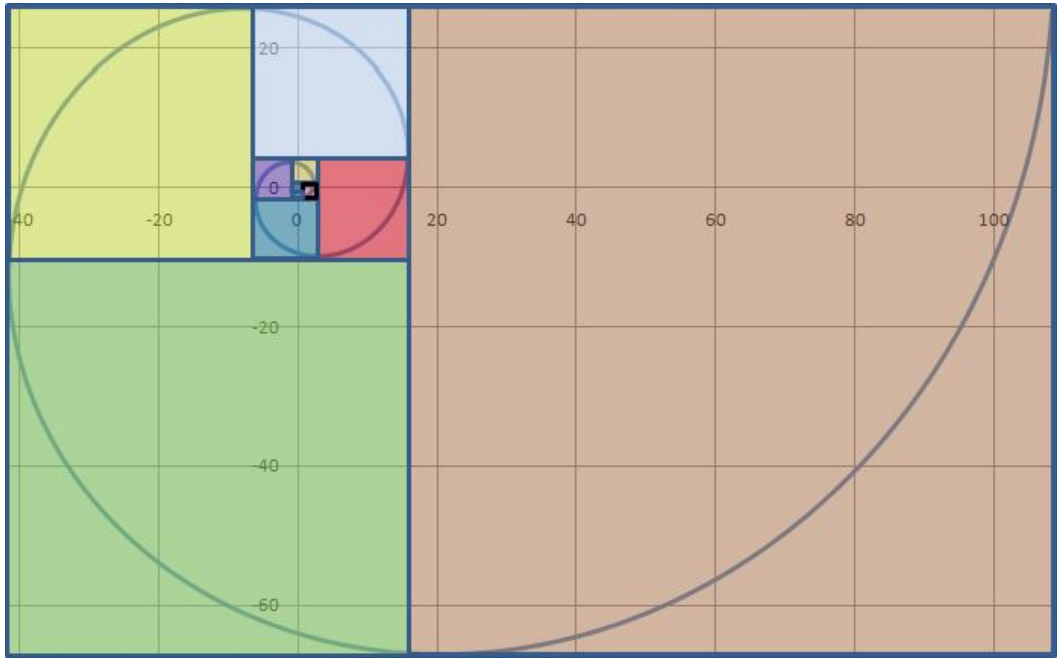
The Set	Computable?	Why?
$\mathbb{R}$	No!	There are uncountably many reals!
$\mathbb{Z}$	Yes!	The set of integers is countable!

Ok Z is Computable. What is Easier to Compute?

$n = 2^{10,000,000,000}$ *or* A random number with 10,000 digits

- Actually, n is easier to compute. But why?
- The answer is found in the field of Kolmogorov complexity.

Which is Easier to Compute? A Different Perspective.



or

The K-Complexity (the smallest possible program) that represents the figure on the left is much smaller than the one on the right.

Algorithmic Construction, A Technical Aside

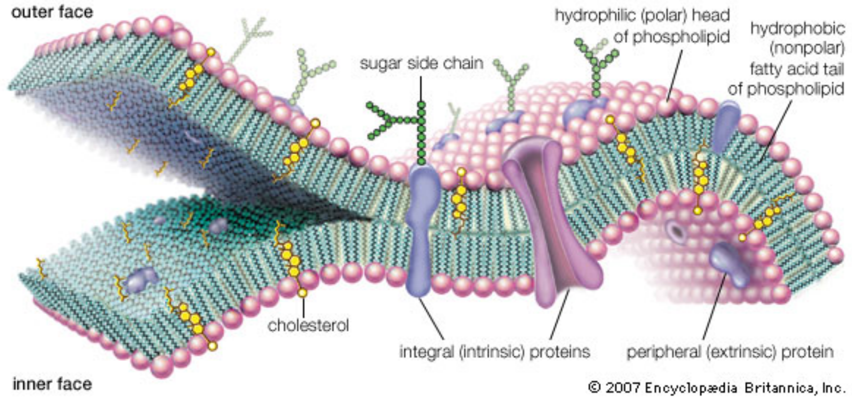
- In the pictures on the previous slide you were mislead a tiny bit.

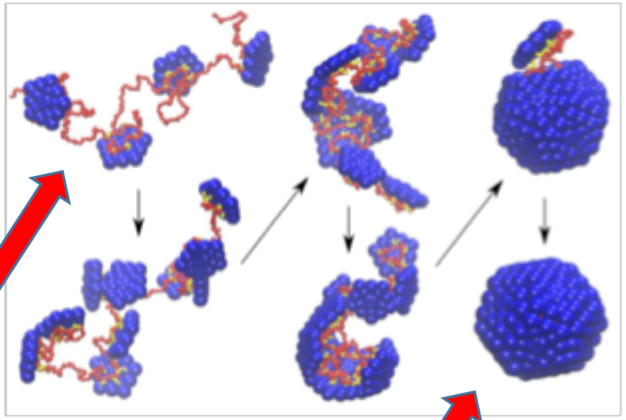
- Any program that, in order to compute a result is the same size as the thing itself is not actually “computing” anything.

  - In the context of Self Assembly, “algorithmic” essentially requires that the “program” for generating the final object is smaller than the final object.

So Then, Why Do This?

- Think of the second figure as a very long program that computes some integer.
- Clearly, this number is computable, but a program that computes it may be intractably large.
  - DNA is a highly desirable computational medium.
  - A single strand of human DNA contains approximately 3 billion base pairs
  - Data density of 1 million Gbits / square inch
  - Each strand is roughly equivalent to an entire processor.
  - …also, we can make really small things!

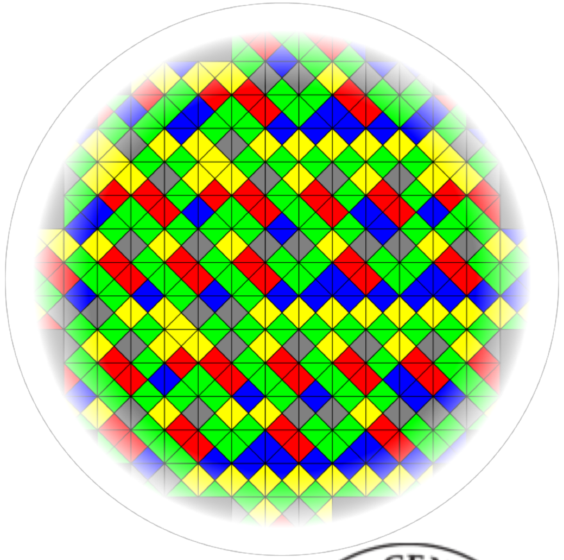




"Less Complex"

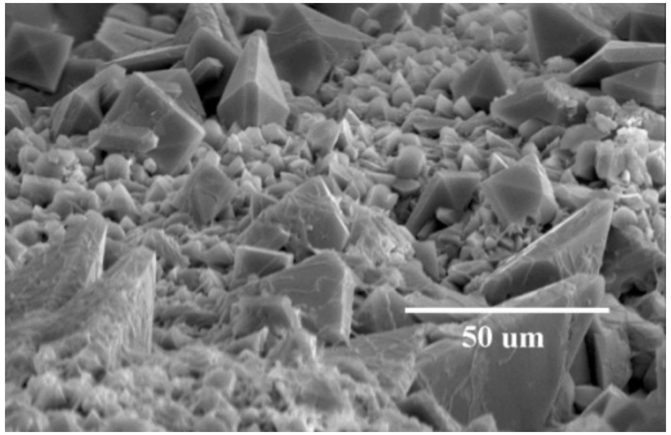
"More Complex"

Models of Self-Assembly



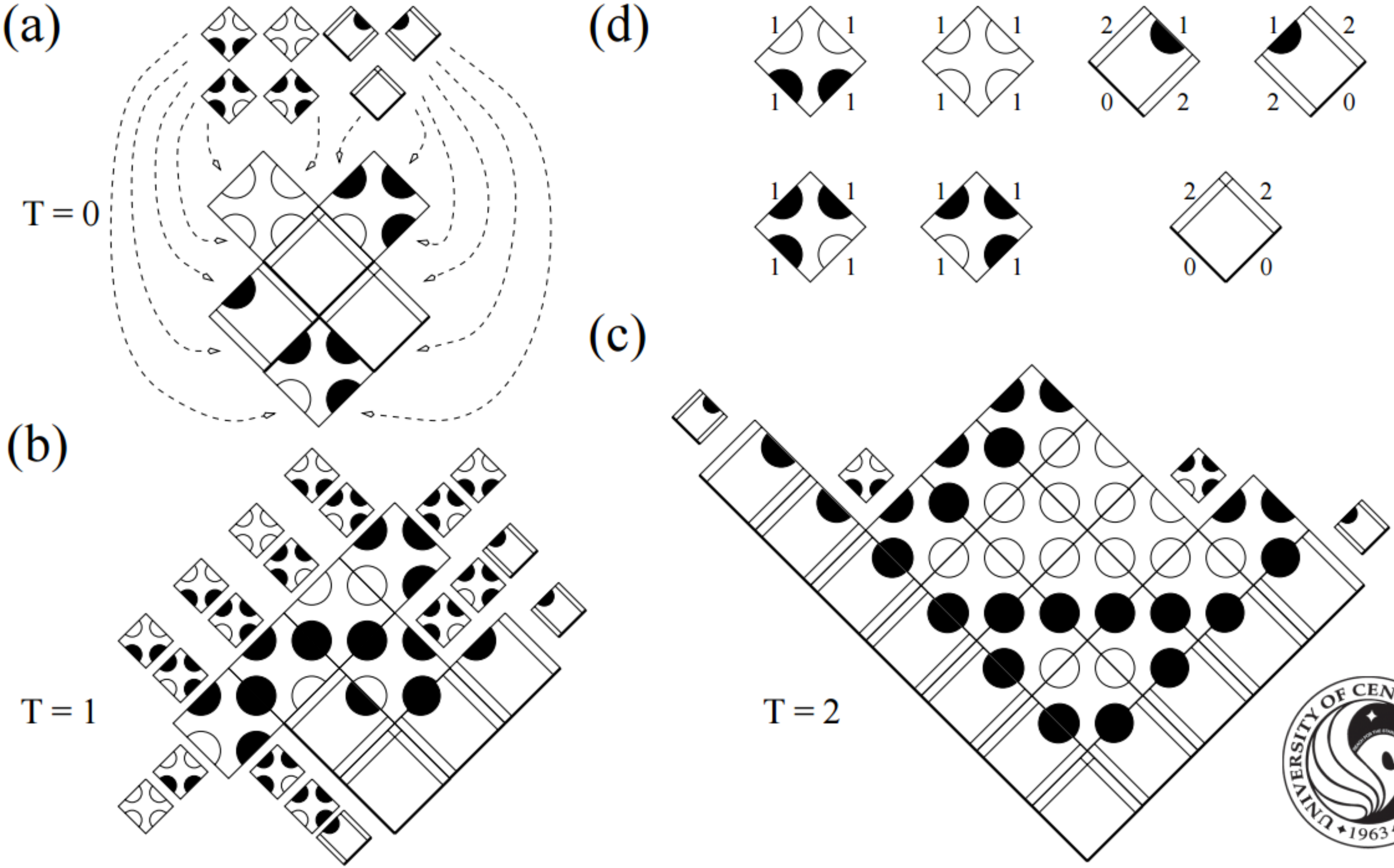
- Many different models of self-assembly exist; the most important were pioneered in Winfree's Doctoral dissertation.
- High level conclusion is that ligation (linking) and annealing (combining base pairs) are enough for computation.
- Two major models of computation, both of which are generalizations of Wang tiles
  - Abstract Tile Assembly Model (aTAM)
  - Kinetic Tile Assembly Model (kTAM)

Abstract Tile Assembly Model



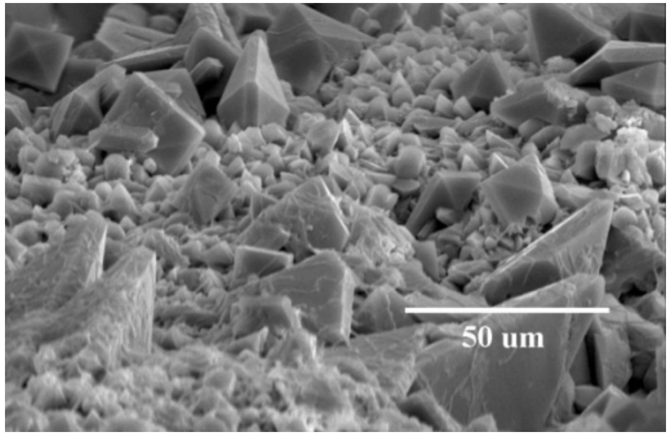
- Main Idea: Turing Universal computation via crystallization

- System has 3 controls:
  - Boundary tiles (allowed to go along the edges)
  - “Rule” tiles, marked tiles encoded with numbers and patterns
  - Temperature, which is an abstraction that controls which tiles may bind with which tiles.


(a)
T = 0
(b)
T = 1
(d)
1 1 1 1
1 1 1 1
2 1 1 2
0 2 2 0
1 1 1 1
1 1 1 1
2 2
0 0
(c)
T = 2

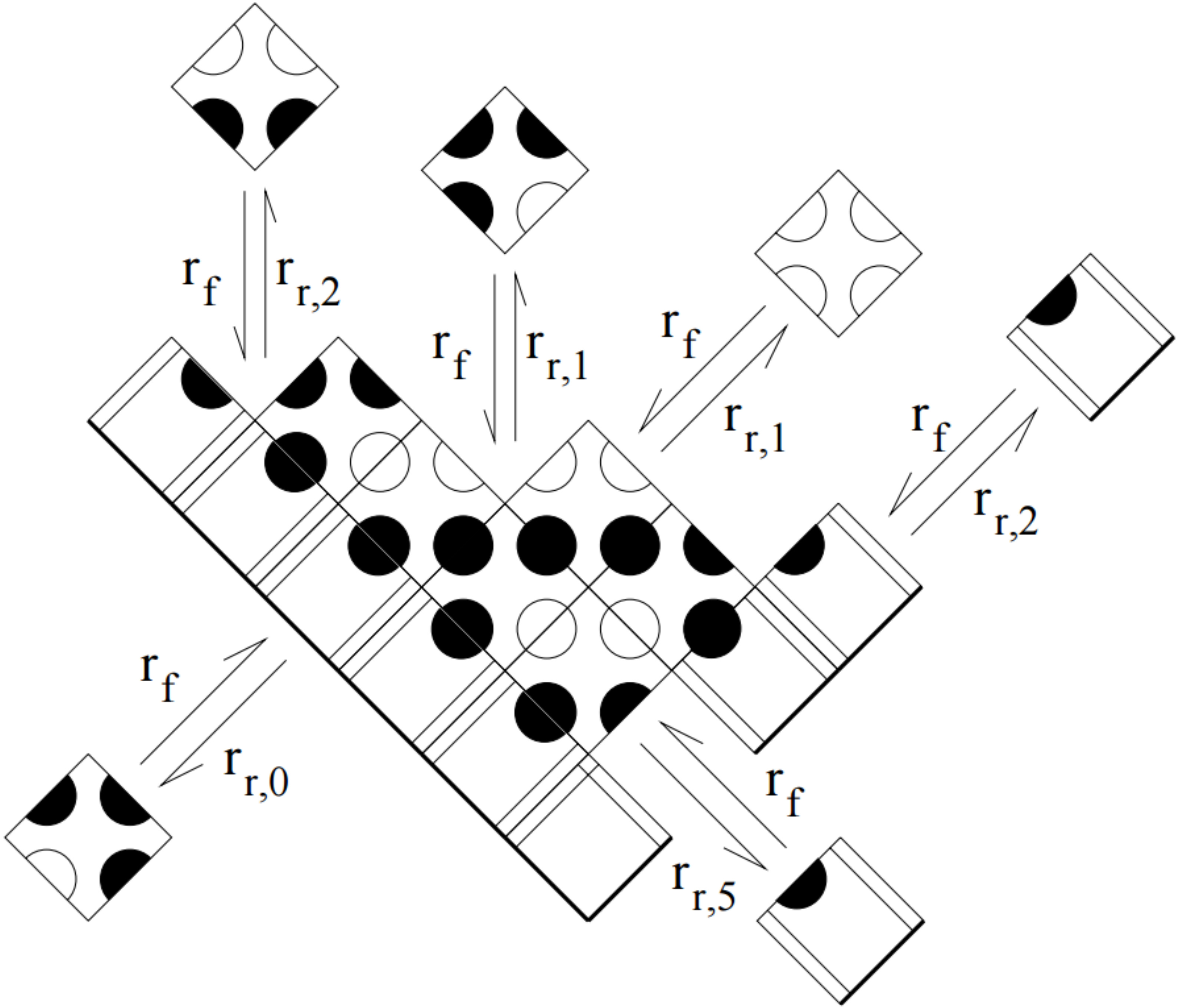
The Sierpinski Triangle

Kinetic Tile Assembly Model



- Though elegant, aTAM is not a realistic model of computation for DNA.
  - aTAM assumes no tile may be lost; in reality this may happen frequently.

- kTAM adds the following:
  - Allows for errors in assembly; provides a mechanism to correct them.
  - Allows tiles to “detach” from structure
  - More accurately models wet-lab assembly of structures.



UNIVERSITY OF CENTRAL FLORIDA ✦ 1963 ✦

Demonstration

Conclusions

- Algorithmic self-assembly, though quite young, is a branch of study that fuses together computer science, biology.

- In this talk we presented an introduction to the field of algorithmic self-assembly:
  - Motivated the study of this very young field.
  - Provided descriptions of two models of computation: aTAM and kTAM.
  - Showed a simulation detailing the computational model presented in this talk using xgrow.

References

- Doty, D. (2012). Theory of algorithmic self-assembly. *Communications of the ACM*, *55*(12), 78-88.
- Winfree, E. (1998). Algorithmic self-assembly of DNA (Doctoral dissertation, California Institute of Technology).
- Li, M., & Vitányi, P. M. (2009). *An introduction to Kolmogorov complexity and its applications*. Springer Science & Business Media.
- Pandey, B. (2007). Bioinformatics. *APH Publishing* Corporation.