Construction of Geometric Structure by Oritatami System

Yo-Sub Han[1] Hwee Kim[2*]

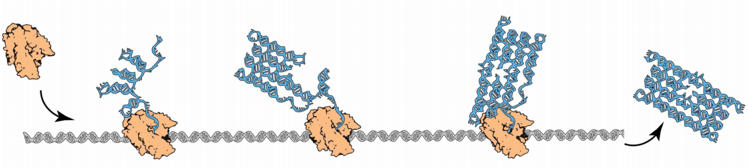
[1]Yonsei University

[2]University of South Florida

The 24th International Conference on DNA Computing and Molecular Programming

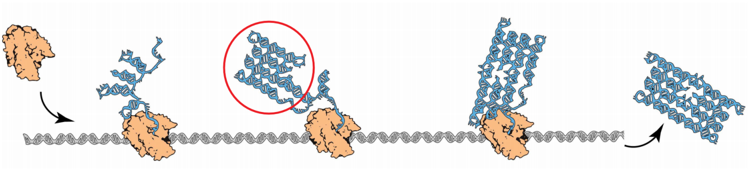
RNA Origami to Oritatami System (OS)

RNA Origami (Geary et al. (2014))



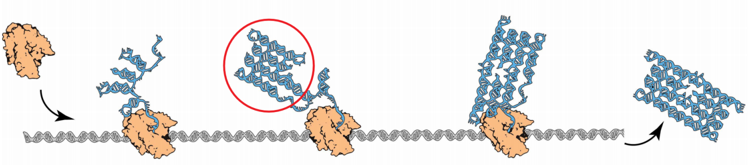
RNA Origami to Oritatami System (OS)

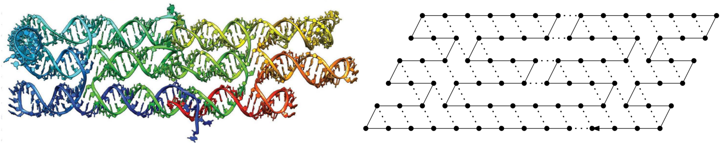
Cotranscriptional folding—folding occurs during transcription



RNA Origami to Oritatami System (OS)

Oritatami System is a mathematical model of computation by cotranscriptional folding (*oritatami* means folding in Japanese).

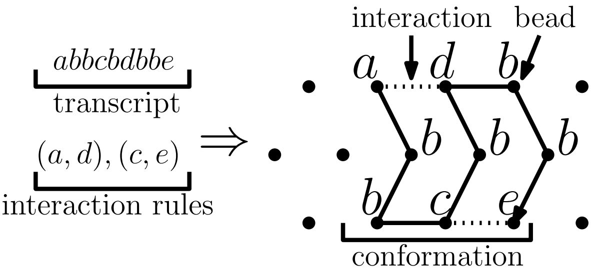




(Left) 3D Image of a tile generated by RNA origami (Right) Conformation that represents the tile

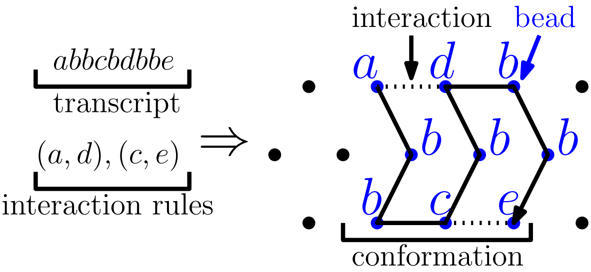
RNA Origami to Oritatami System (OS)

RNA Origami	Oritatami System
Nucleotides	Beads
Transcript	Sequence of beads connected by a line
h-bonds between nucleotides	Interactions
Cotranscriptional folding rate	Delay
Resulting secondary structure	Conformation



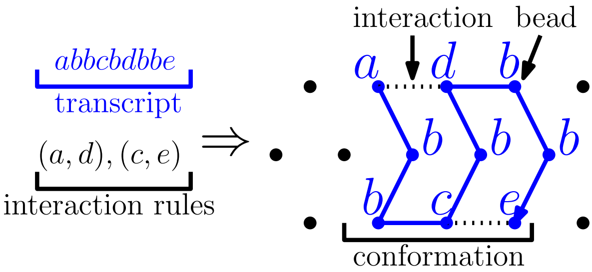
RNA Origami to Oritatami System (OS)

RNA Origami	Oritatami System
Nucleotides	Beads
Transcript	Sequence of beads connected by a line
h-bonds between nucleotides	Interactions
Cotranscriptional folding rate	Delay
Resulting secondary structure	Conformation



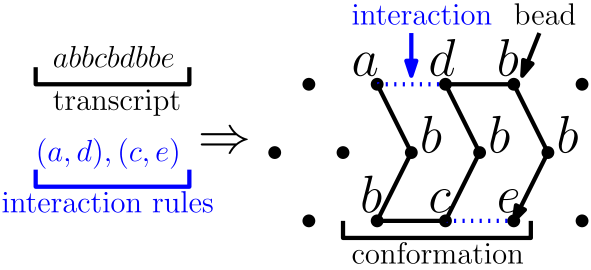
RNA Origami to Oritatami System (OS)

RNA Origami	Oritatami System
Nucleotides	Beads
Transcript	Sequence of beads connected by a line
h-bonds between nucleotides	Interactions
Cotranscriptional folding rate	Delay
Resulting secondary structure	Conformation



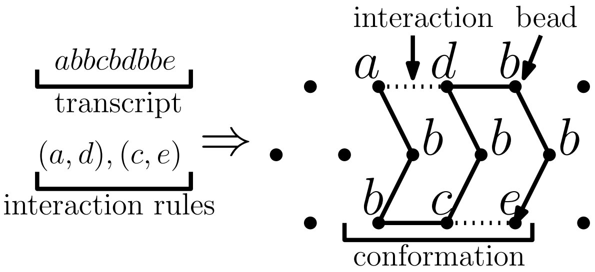
RNA Origami to Oritatami System (OS)

RNA Origami	Oritatami System
Nucleotides	Beads
Transcript	Sequence of beads connected by a line
h-bonds between nucleotides	Interactions
Cotranscriptional folding rate	Delay
Resulting secondary structure	Conformation



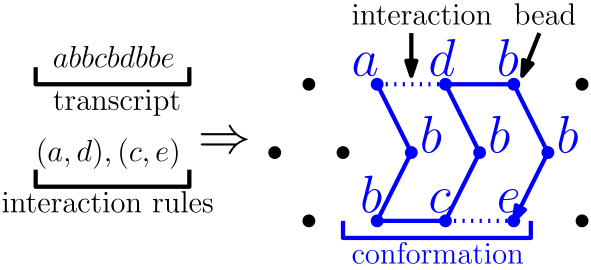
RNA Origami to Oritatami System (OS)

RNA Origami	Oritatami System
Nucleotides	Beads
Transcript	Sequence of beads connected by a line
h-bonds between nucleotides	Interactions
Cotranscriptional folding rate	Delay
Resulting secondary structure	Conformation



RNA Origami to Oritatami System (OS)

RNA Origami	Oritatami System
Nucleotides	Beads
Transcript	Sequence of beads connected by a line
h-bonds between nucleotides	Interactions
Cotranscriptional folding rate	Delay
Resulting secondary structure	Conformation



Dynamics of OS (Geary et al. (2015))

The seed C_σ is the initial conformation. We stabilize each bead of the transcript w as follows:

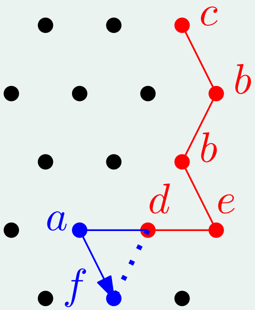
1. The bead look ahead up to next δ (delay) beads
2. Following rules in the ruleset $\mathcal{H}$, each pair of adjacent beads can form an interaction
3. The arity α denotes the maximum number of interactions that a bead can form
4. The first bead stabilizes as to maximize the number of interactions that the lookahead forms

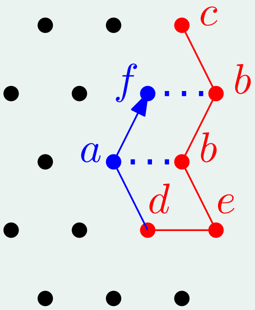
Dynamics of OS (Geary et al. (2015))

Example

The delay of the system is 2. The seed is given as the red line. The transcript is *afe*. The ruleset is $\{(a, b), (b, f), (d, f), (d, e)\}$.

$$w = \underline{af}e$$



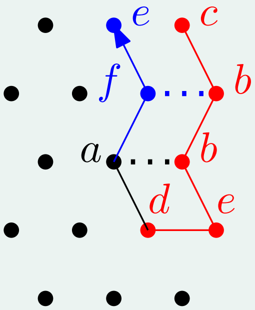


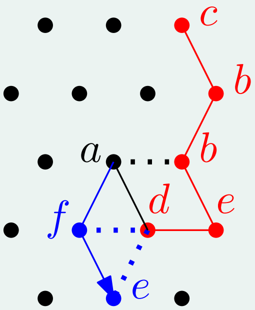
Dynamics of OS (Geary et al. (2015))

Example

The delay of the system is 2. The seed is given as the red line. The transcript is *afe*. The ruleset is $\{(a,b),(b,f),(d,f),(d,e)\}$.

$$w = a\underline{fe}$$



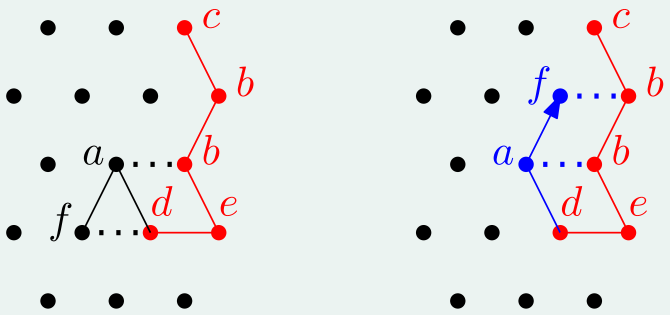


Dynamics of OS (Geary et al. (2015))

Example

The delay of the system is 2. The seed is given as the red line. The transcript is *afe*. The ruleset is $\{(a,b),(b,f),(d,f),(d,e)\}$.

$$w = a\underline{fe}$$

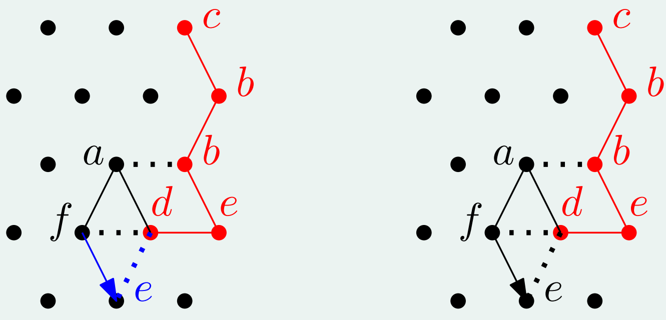


Dynamics of OS (Geary et al. (2015))

Example

The delay of the system is 2. The seed is given as the red line. The transcript is *afe*. The ruleset is $\{(a,b),(b,f),(d,f),(d,e)\}$.

$$w = af\underline{e}$$

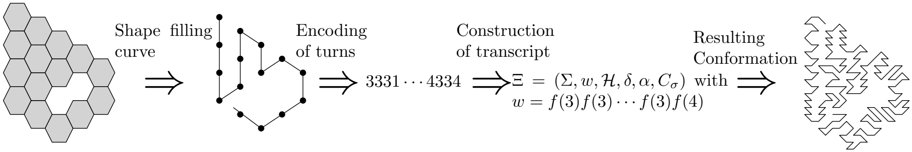


Previous Works

- Design
  - OS is Turing complete (Geary et al. (2015))
  - We can construct a binary counter (Geary et al. (2016))
  - We can construct a tautology checker (and a SAT solver) (Han et al. (2018))
- Hardness and Complexity
  - OS equivalence problem is coNP-hard (Han et al. (2018))
  - In general, it is NP-hard to retrieve a ruleset to fold the given conformation (Ota and Seki (2017))
  - Self-attraction removal by bead type copying (Han et al. (2017))
  - Ruleset optimization problem (Han and Kim (2017))
- Geometric Construction
  - Construction of the dragon curve (Masuda et al. (2018))
  - Hardness of constructing geometric shapes using certain delays (Rogers and Seki (2017))

Overview

- We want to design a geometric structure constructing OS (GEOS)
- The target structure is given as a set of points in an arbitrary lattice
- We want a terminal conformation that corresponds to the grid graph of the given points
- Idea: Design smaller modular OSs (hinges) for every possible pair of adjacent points in the target structure



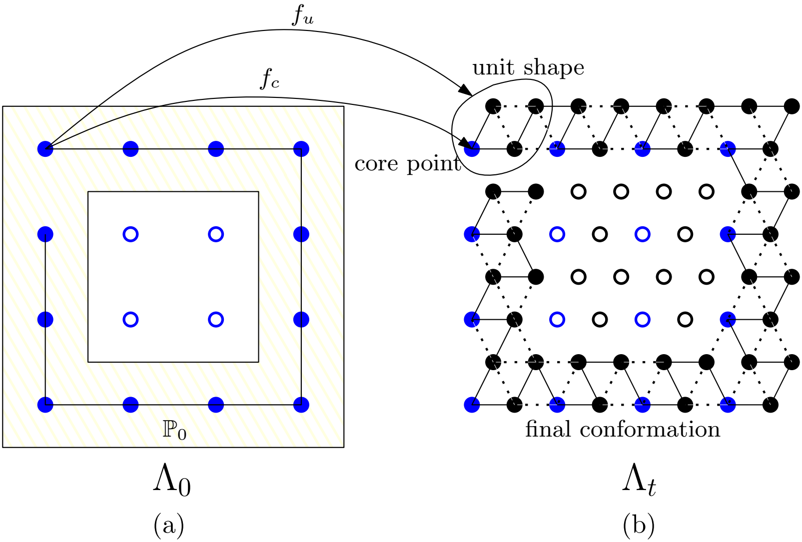
Comparison with Demaine et al. (2018)

	Demaine et al.	Our paper
Target Lattice	Triangular	Triangular/Square
Scale (in our paper's measure)	≥ 19	25 in both OSs
Rigidity	1	$\frac{24}{25}/1$
Filling Mecahnism	Global (Fills the whole structure. No ordering of points.)	Local (Fills each point at a time.)
Delay	1	4 in both OSs
Arity	4	3 in both OSs

Problem Specification

- The input for the GEOS is as follows:
  - A lattice Λ_0 on the plane
  - A shape that we want to fill on the lattice, which is given by the set $\mathbb{P}_0$ of points
- The output should include the follows:
  - A triangular lattice Λ_t that spans Λ_0
  - An injective function f_c that maps $p \in \Lambda_0$ to $p_c \in \Lambda_t$. We call p_c the core point
  - A bijective mapping $f_u : p \in \Lambda_0 \to \mathbb{U}(p) \subset \Lambda_t$ that maps $p \in \Lambda_0$ to $\mathbb{U}(p) \subset \Lambda_t$. We call the induced graph of $\mathbb{U}(p)$ the unit shape, and $|\mathbb{U}(p)|$ the scale (inspired by Doty et al. (2012)) to measure the efficiency of GEOS
  - A deterministic OS Ξ on Λ_t, where the final conformation covers at least one point in $\mathbb{U}(p)$ for each $p \in \mathbb{P}_0$

Problem Specification



Desirable Features (inspired by Soloveichik and Winfree (2007))

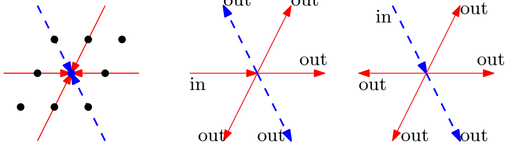
- The scale should be as small as possible
- The number of beads in Ξ should be as small as possible
- The final conformation should fill as many points in the unit shape as possible: We use the rigidity to refer to the lower bound of the ratio of the number of filled points to the scale

Designing a GEOS

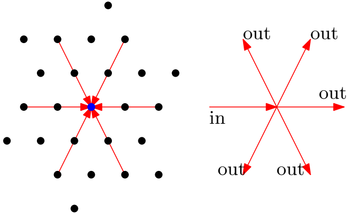
- A space filling curve can be represented by a sequence of unit vectors
- For each pair of adjacent unit vectors, we design a partial OS (hinge) that fills adjacent unit shapes

Design Guidelines

(i) Unit shapes should be identical and symmetric



(a): 10 hinges

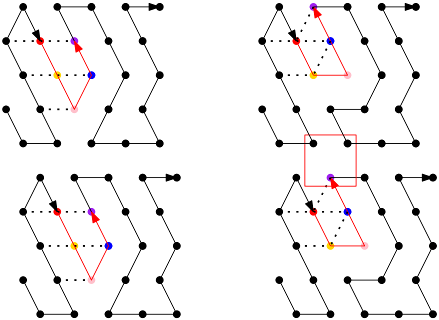


(b): 5 hinges

Design Guidelines

(ii) We use core beads to cover the core point and hinge beads to connect core beads

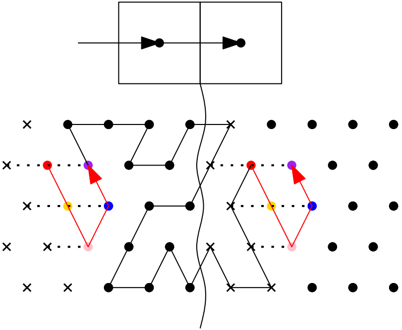
1. Cores should be symmetric
2. Core beads should not be revealed on the surface of the unit shape



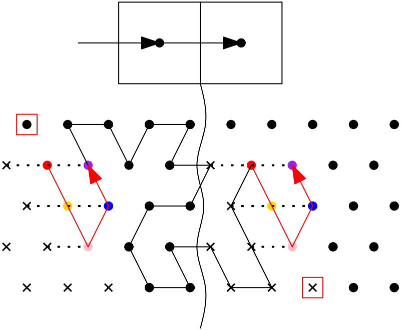
(a) (b)

Design Guidelines

(iii) For points to fill between two cores, there should exist a Hamiltonian path



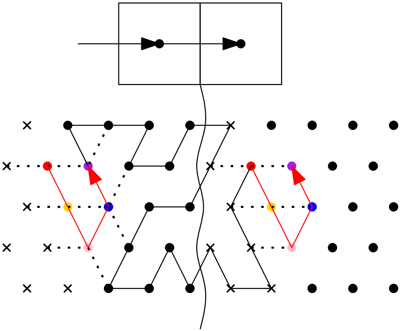
(a)



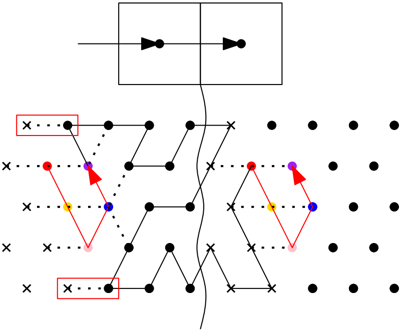
(b)

Design Guidelines

(iv) Hinge beads in different types of hinges should not interact with each other



(a)



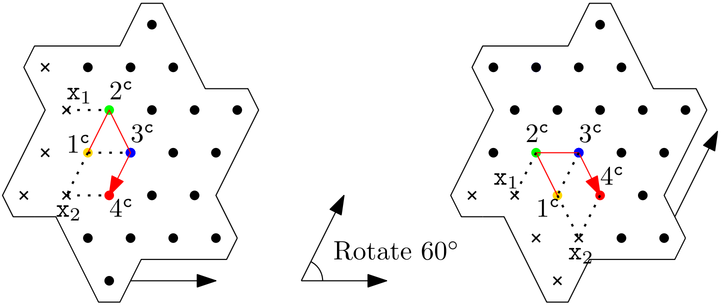
(b)

Design Guidelines

Guidelines	Small $\|\Sigma\|$	High rigidity	Avoiding unintended interactions
(i)	*		
(ii)(a)	*		
(ii)(b)			*
(iii)		*	
(iv)			*

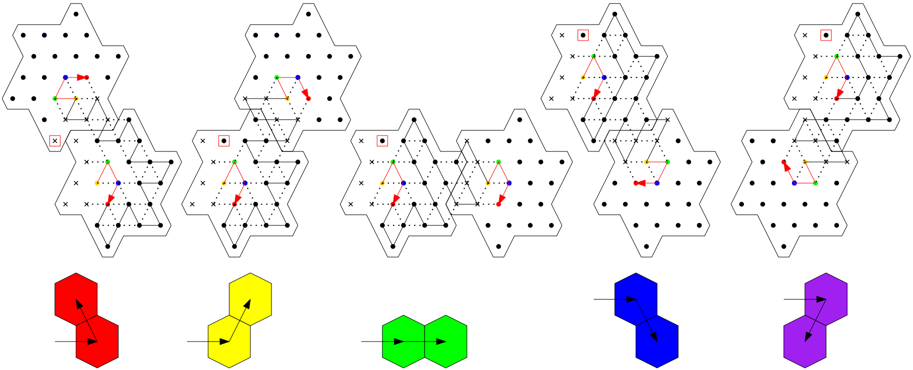
Summary of strengths of four guidelines on desirable features of GEOS.

$\Xi_\triangle$ from a Triangular Lattice



Core beads of $\Xi_\triangle$

$\Xi_{\triangle}$ from a Triangular Lattice

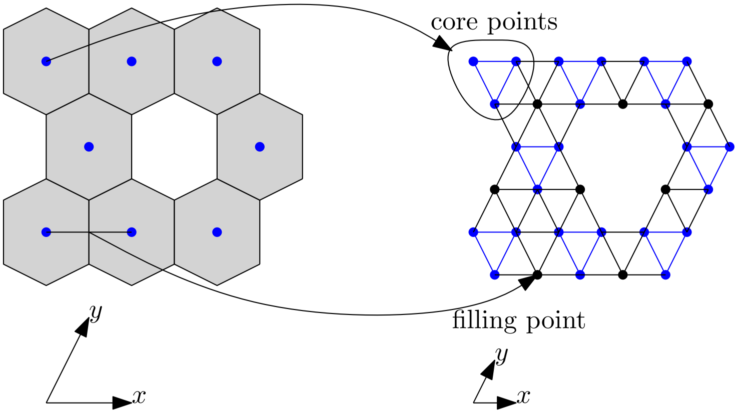


Hinges of $\Xi_{\triangle}$

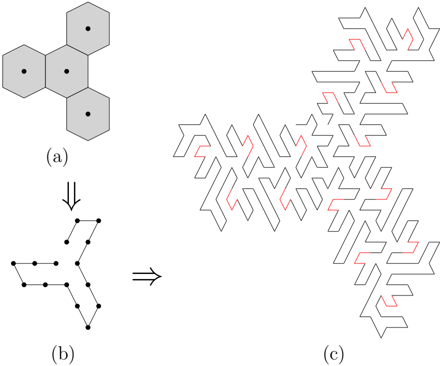
- Following all guidelines except (iii)
- Scale=25 (75 when group three points), Rigidity=24/25
- 5 hinges, 104 Bead types

$\Xi_\triangle$ from a Triangular Lattice

- Not all connected triangular grid graph is Hamiltonian
- Solution: Make the graph locally connected, for exchange of tripling the scale

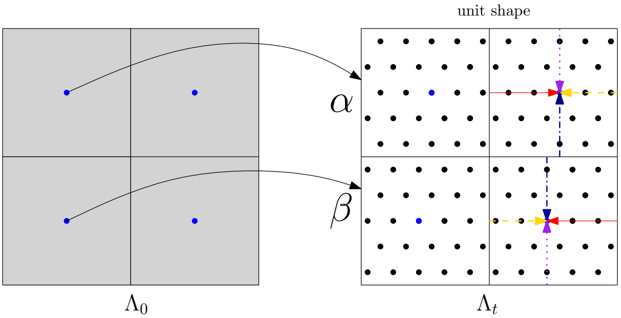


$\Xi_\triangle$ from a Triangular Lattice



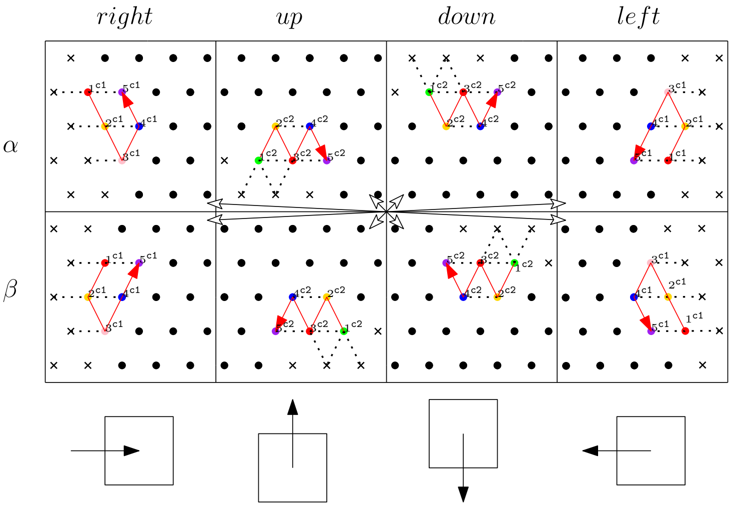
An example of $\Xi_\triangle$

$\Xi_\square$ from a Square Lattice



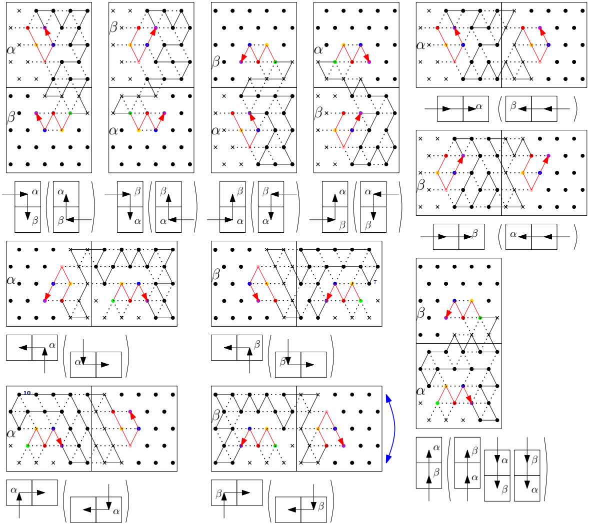
- Two different (reflectional symmetric) unit shapes α and β
- Scale=25
- Rigidity=1
- 11 hinges
- 223 Bead types
- At one hinge, it requires horizontal rotation of the succeeding hinges

$\Xi_{\square}$ from a Square Lattice



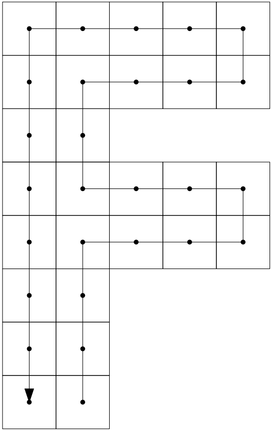
Core beads of $\Xi_{\square}$

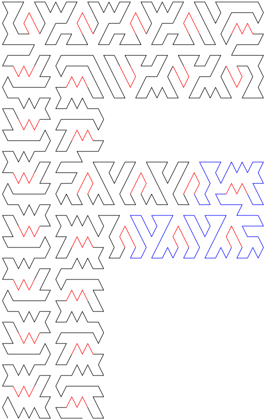
$\Xi_\square$ from a Square Lattice



Hinges of $\Xi_\square$

$\Xi_\square$ from a Square Lattice

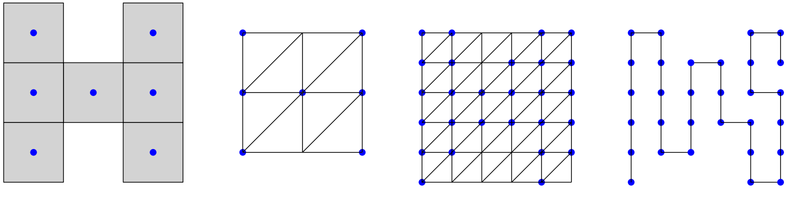




Core beads are colored in red. Horizontally rotated hinges are colored in blue

$\Xi_\square$ from a Square Lattice

- Finding a Hamiltonian path from a square grid graph is NP-complete
- Solution: transform a square grid into a triangular grid
- Scale quadruples to 100



(a) (b) (c) (d)

(a) The input shape (b) The input shape on the affine triangular grid (c) Construction of a locally connected graph (d) A retrieved path

Conclusions

- Oritatami System (OS) is a mathematical model of computation by cotranscriptional folding
- We proposed the generalized OS (GEOS) design guideline to fill the given geometric structure
- We proposed two GEOSs according to guidelines
- Future works: Optimization. Finding bounds for number of bead types.

GEOS	Scale	Rigidity	# of Hinges	# of Beads
$\Xi_{\triangle}$	25 (75 when group three core points)	24/25	5	104
$\Xi_{\square}$	25 (100 when group four core points)	1	11	223

Summary of two proposed GEOSs

Thank You!

References I

Demaine, E. D., Hendricks, J., Olsen, M., Patitz, M. J., Rogers, T. A., Schabanel, N., Seki, S., and Thomas, H. (2018). Know when to fold 'em: Self-assembly of shapes by folding in oritatami. In *Proceedings of the 24th International Conference on DNA Computing and Molecular Programming*. To appear.

Doty, D., Lutz, J. H., Patitz, M. J., Schweller, R. T., Summers, S. M., and Woods, D. (2012). The tile assembly model is intrinsically universal. In *Proceedings of the IEEE 53rd Annual Symposium on Foundations of Computer Science*, pages 302–310.

Geary, C., Meunier, P., Schabanel, N., and Seki, S. (2015). Efficient universal computation by greedy molecular folding. *CoRR*, abs/1508.00510.

References II

Geary, C., Rothemund, P. W. K., and Andersen, E. S. (2014). A single-stranded architecture for cotranscriptional folding of RNA nanostructures. *Science*, 345:799–804.

Geary, C. W., Meunier, P., Schabanel, N., and Seki, S. (2016). Programming biomolecules that fold greedily during transcription. In *Proceedings of the 41st International Symposium on Mathematical Foundations of Computer Science*, pages 43:1–43:14.

Han, Y. and Kim, H. (2017). Ruleset optimization on isomorphic oritatami systems. In *Proceedings of the 23rd International Conference on DNA Computing and Molecular Programming*, pages 33–45.

Han, Y., Kim, H., Ota, M., and Seki, S. (2018). Nondeterministic seedless oritatami systems and hardness of testing their equivalence. *Natural Computing*, 17(1):67–79.

References III

Han, Y., Kim, H., Rogers, T. A., and Seki, S. (2017). Self-attraction removal from oritatami systems. In *Proceedings of the 19th International Conference on Descriptional Complexity of Formal Systems*, pages 164–176.

Masuda, Y., Seki, S., and Ubukata, Y. (2018). Towards the algorithmic molecular self-assembly of fractals by cotranscriptional folding. In *Proceedings of the 23rd International Conference on Implementation and Application of Automata*, pages 261–273.

Ota, M. and Seki, S. (2017). Rule set design problems for oritatami system. *Theoretical Computer Science*, 671:16–35.

Rogers, T. A. and Seki, S. (2017). Oritatami system; a survey and the impossibility of simple simulation at small delays. *Fundamenta Informaticae*, 154(1-4):359–372.

Soloveichik, D. and Winfree, E. (2007). Complexity of self-assembled shapes. *SIAM Journal of Computing*, 36(6):1544–1569.