

ASSEMBLABLE

INTERLOCKING

POLYOMINOES



HELLO!

I am Andres Bejarano

Interested in Computer Graphics, Computational
Geometry and Algorithms and Theory

Currently working with professor Christoph Hoffmann



Brief background about Polyominoes

Finding Assemblable Interlocking Polyominoes

Current results

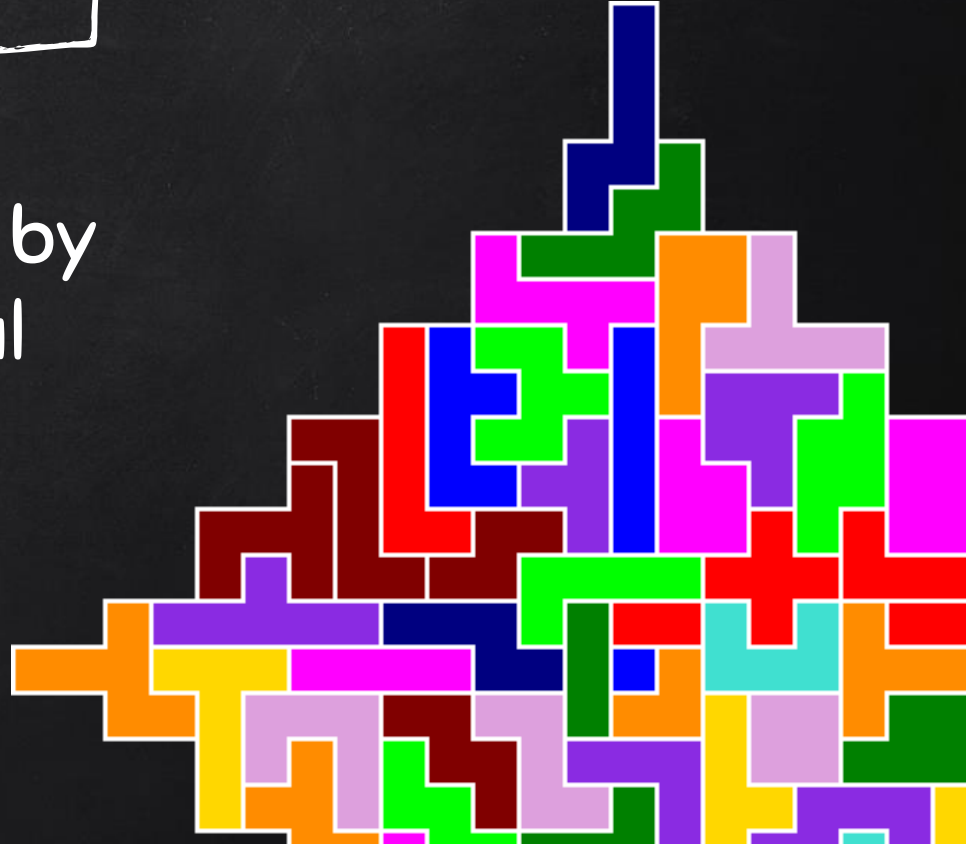
Working ideas

BRIEF BACKGROUND ABOUT POLYOMINOES

Let's start with the basic stuff

“

“A polyomino is a plane geometric figure formed by joining one or more equal squares edge to edge.”



SOME HISTORY

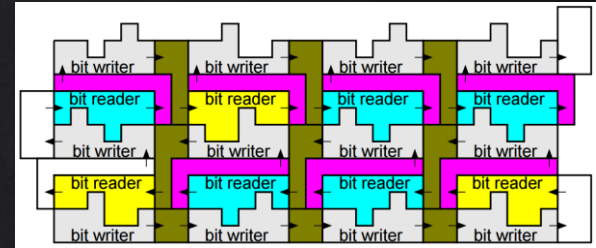
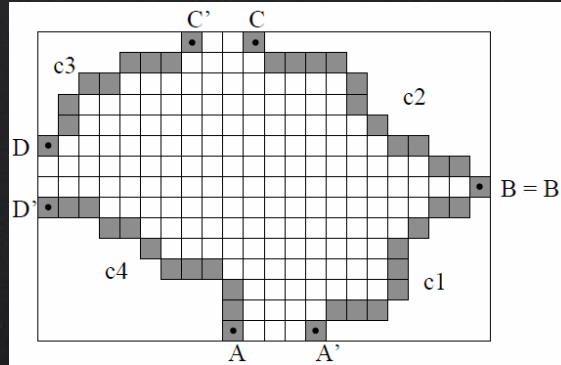
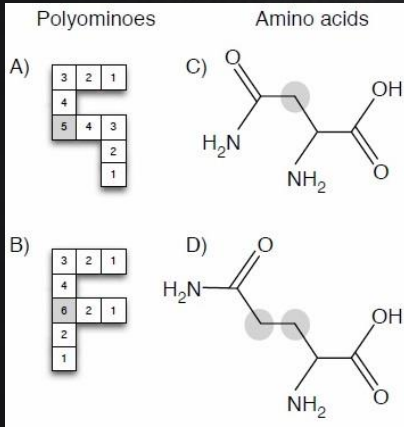
First problem published in 1907 by Henry Ernest Dudeney.

Solomon Golomb named them in 1953.

Martin Gardner popularized them in Scientific American.

Now everyone is talking about them... Well, everyone interested.

SOME POLYOMINOES IN REAL LIFE (AKA NOT VIDEO GAMES)





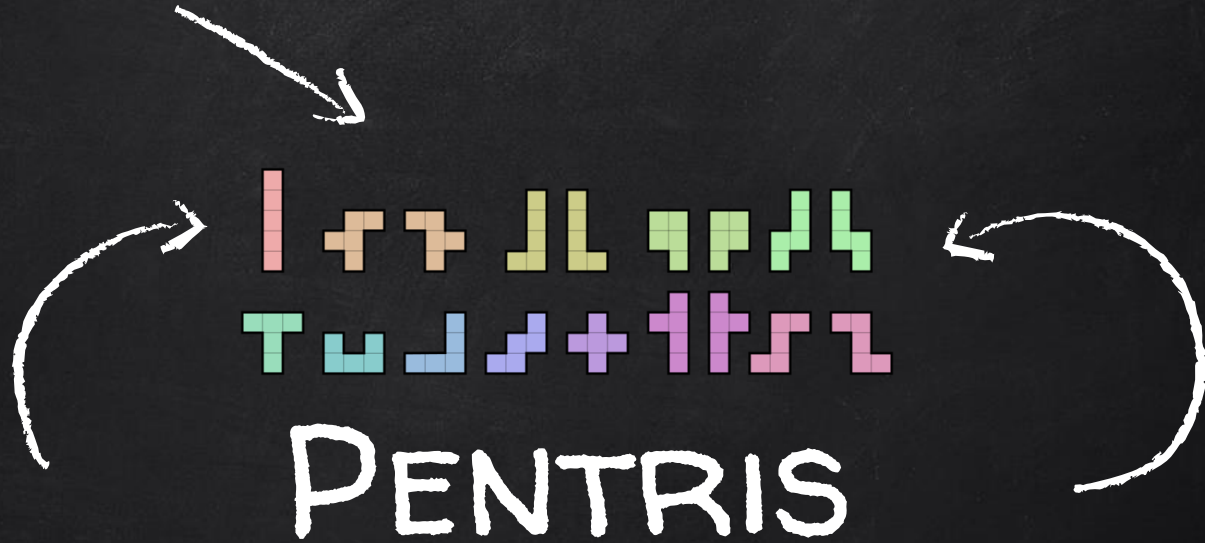
TETRIS: BEST POLYOMINO APPLICATION KNOWN SO FAR



Thanks to Alexey Pajitnov.

One of the most popular
videogames in history.

Computationally speaking, its
complexity is NP-Complete.



Have you played it?
Do you think it is easier than Tetris?
Good luck!



TRADITIONAL RESEARCH WITH/USING POLYOMINOES

Plane Tessellation

Puzzles commonly ask for tiling a given region with a given set of polyominoes, such as the 12 pentominoes.

Counting Polyominoes

No formula has been found except for special classes of polyominoes. A number of estimates are known, and there are algorithms for calculating them.

Discrete Tomographies

Useful for the reconstruction of two dimensional objects from their two orthogonal projections.

Protein Folding (poly GP map)

The mix of a genotype and a self-assembly process on a square lattice leads to the formation of phenotypes with different square tile building blocks conjoined along interacting edges.

Self-Assembly (polyTAM)

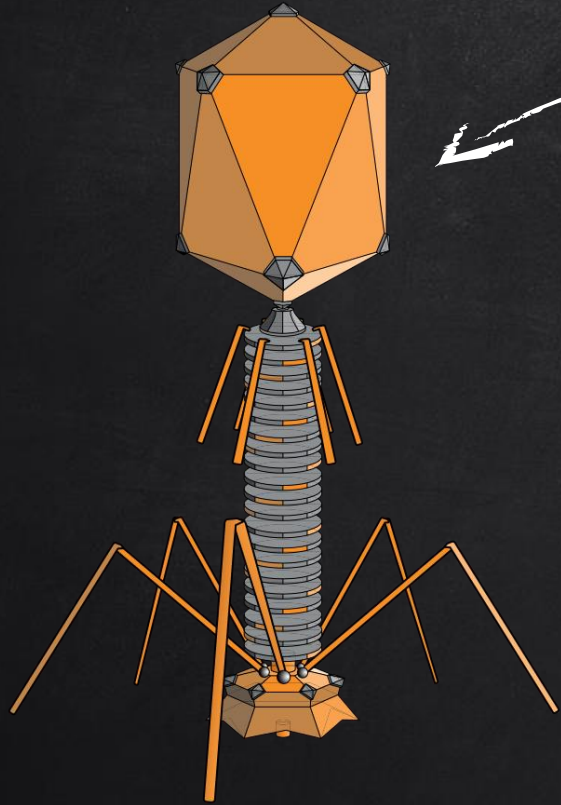
Polyominoes have enough geometric complexity to allow a polyTAM system at temperature 1, composed only of tiles of that shape, to perform Turing universal computation.

Interlocking Properties

A system of grid polygons such that the pieces interlock themselves (no piece can be moved far away from the rest).

FINDING ASSEMBLABLE INTERLOCKING POLYOMINOES

What you need to know in a nutshell

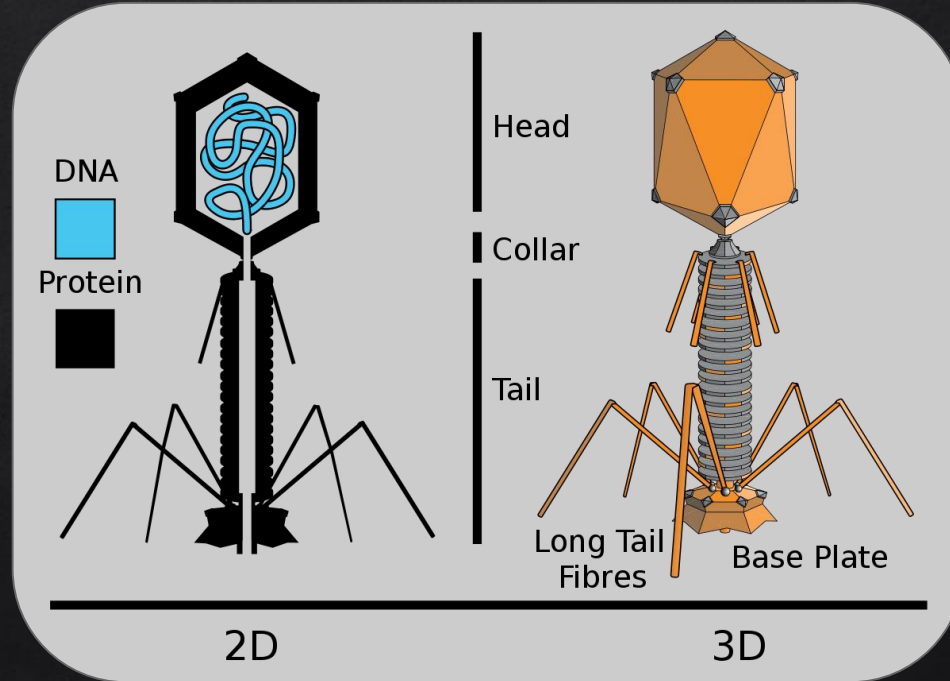


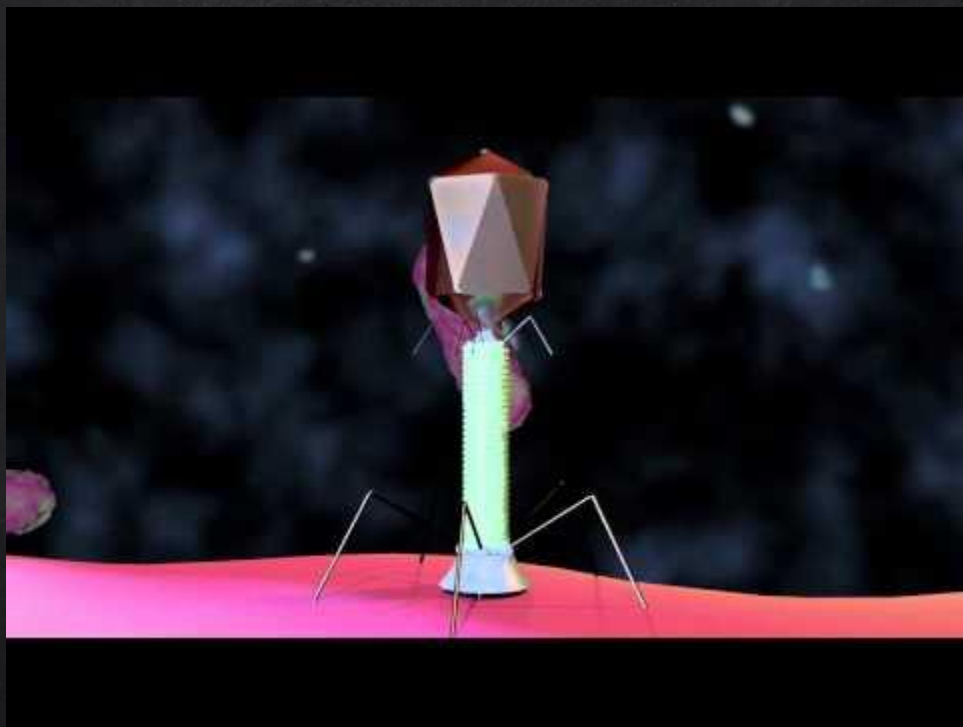
DO YOU KNOW WHAT'S THE STRUCTURE
OF THE LEFT?

- a) A self-deployed non-used lunar module prototype
- b) A proposed antenna for long-distance communications
- c) The structure of a myovirus bacteriophage
- d) The next lovely robot for Star Wars Episode VIII



THE STRUCTURE OF A TYPICAL MYOVIRUS BACTERIOPHAGE



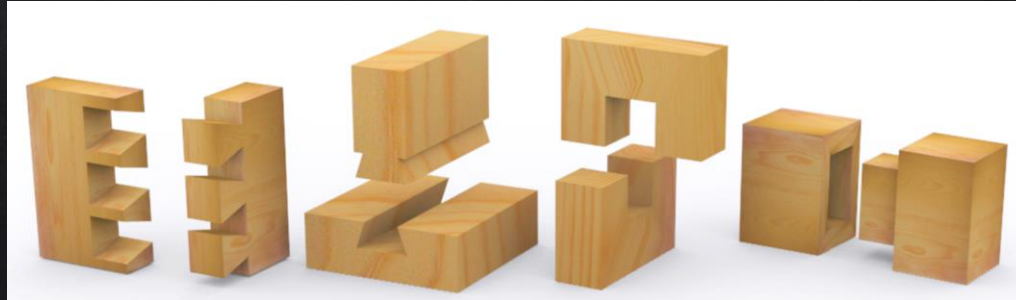




“So, the big problem that biologists and virologists are facing is to understand how those building blocks get together to form the container.”

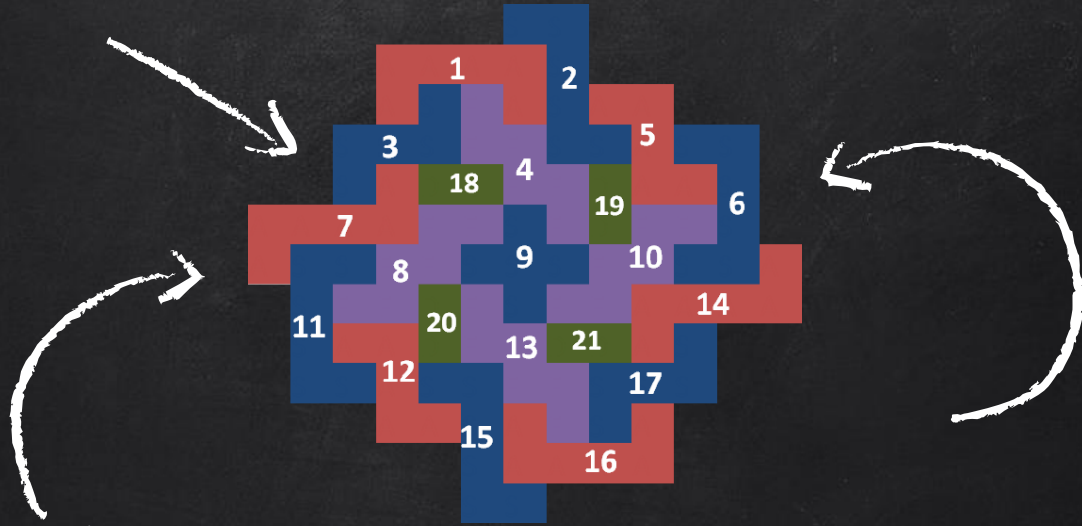
Reidun Twarock – Department of Mathematics at the University of York.

RECENT ADVANCE IN FURNITURE DESIGN





“A system of polyominoes is collectively interlocked if no polyominoes can be moved arbitrarily far away [in the plane] from any of the others.”



AN INTERLOCKED SET

Skeptical? Let's see the demo!



NOT A SIMPLE TASK, IT'S JUST INTRACTABLE

It is PSPACE Hard!

The Totally Quantified Boolean Formula (TQBF) problem, which is PSPACE Complete, can be reduced to the sliding blocks problem.

Determining interlockedness is PSPACE hard for a system of polyominoes with only hexominoes and smaller polyominoes.

What is PSPACE?

It is the class of all languages recognizable by polynomial space bounded Deterministic Turing Machine programs that halt on all inputs.

It is still unknown whether there exist problems solvable in polynomial space that cannot be solved in polynomial time. (is $P = PSPACE$?)

Is it always PSPACE Hard?

It is if we want to tessellate a complete region using hexominoes and smaller polyominoes.

But sometimes we don't want to interlock all the region...



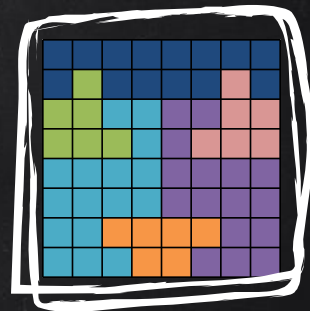
LESSONS FROM OUR PREVIOUS WORK



Space filling curves

Full tessellation of the lattice

Key pieces are too big



We have to do better

Maybe we do not require all the interlockings

Still “assemblable”



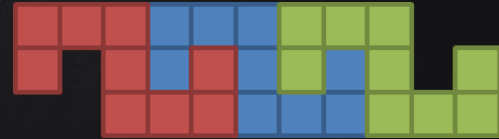
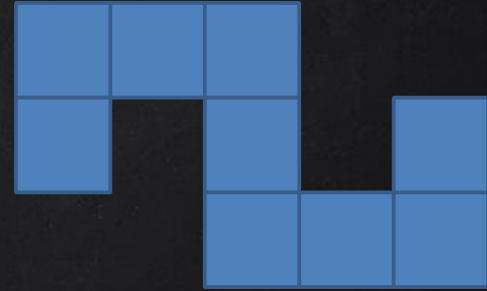
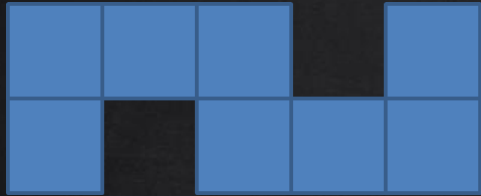


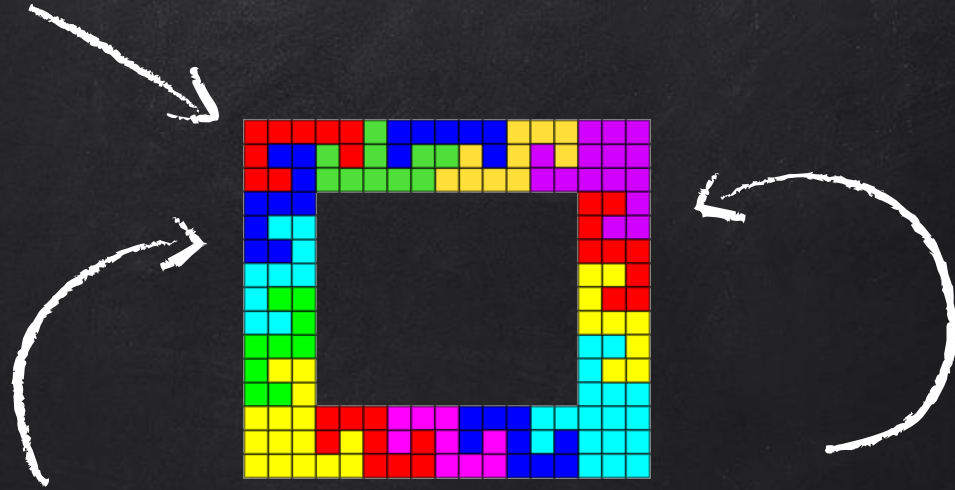
THE GREEKS WERE RIGHT!





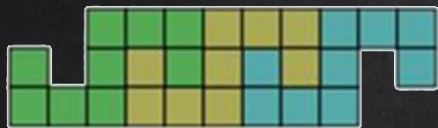
OUR APPROACH: S-SHAPED PIECES





ASSEMBLABLE SETS

Skeptical again? Let's return back to the demo!

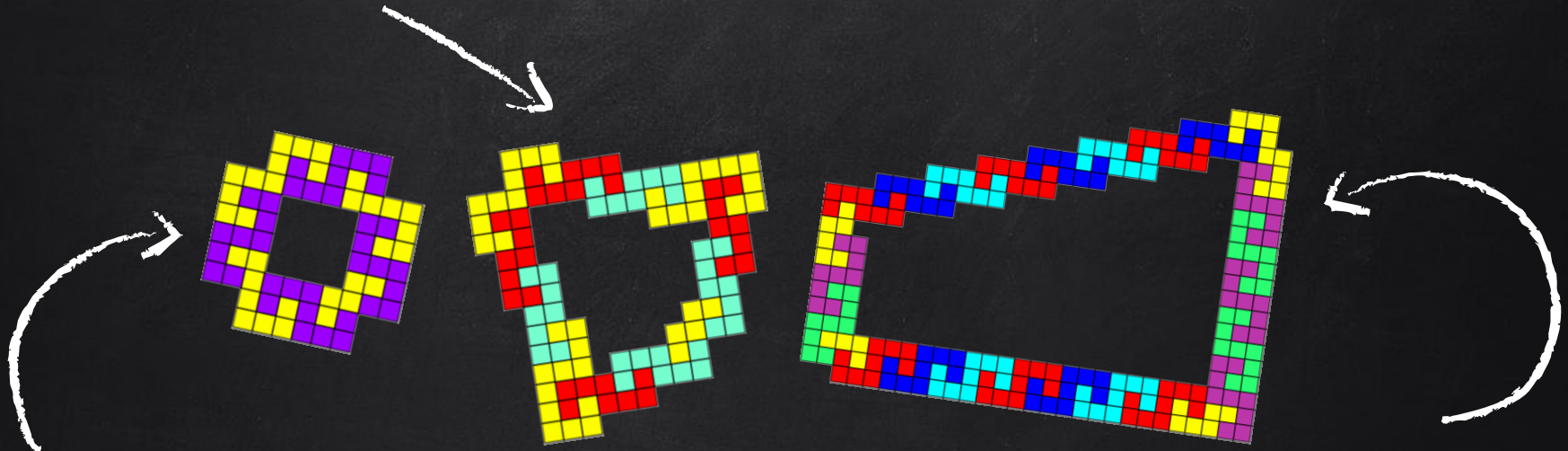


HOW TO MAKE IT ASSEMBLABLE?

1. Find three consecutive s-shaped pieces
2. Merge them
3. Cut the blue piece (key piece!)
4. Make a step cut below the key piece following the direction of the original merged s-shaped pieces.

CURRENT RESULTS

We have some interesting sets to show



CLOVERS, FOXGLOVES AND STAIRCASES

No more doubts! The demo has the answer

WORKING IDEAS

And some other tentative thoughts as well



DEGREES OF INTERLOCKING POLYOMINOES

Is it always PSPACE Hard?

It is when using hexominoes and smaller polyominoes.

Still not known how complex it is when using additional help (i.e., glueing) or some predefined pieces (i.e., s-shaped pieces).

Example: k -glues interlocking class

Let's say we allow a k number of glues between squares in the lattice.

- Boundaries? (i.e., min k ?)
- Algorithm?
- Is the solution always an interlocking set?
- Running time?
- Can we do better/generalize?



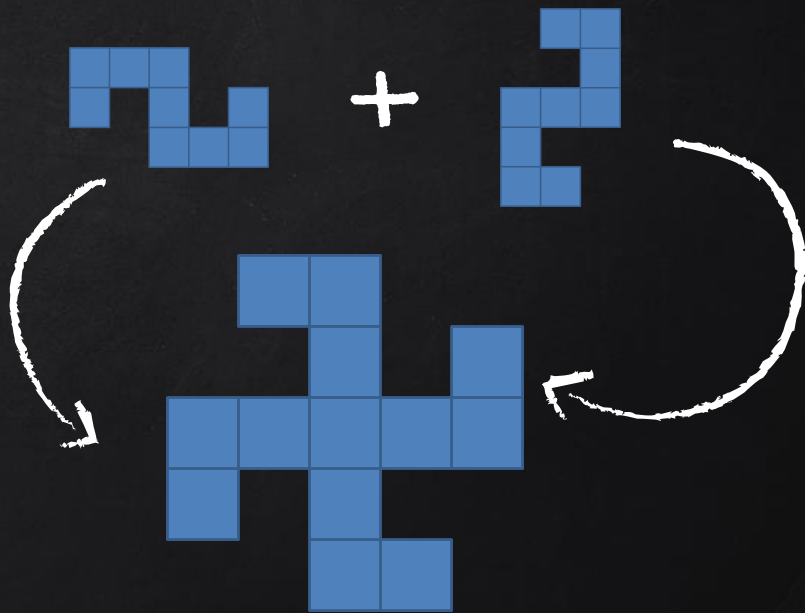
HOW ABOUT MORE GENERAL INTERLOCKING PIECES?

Goal: Interlock everything!

S-shaped pieces are good for linear segments.

Idea: replicate the design in the four directions.

Not a popular polyomino though.





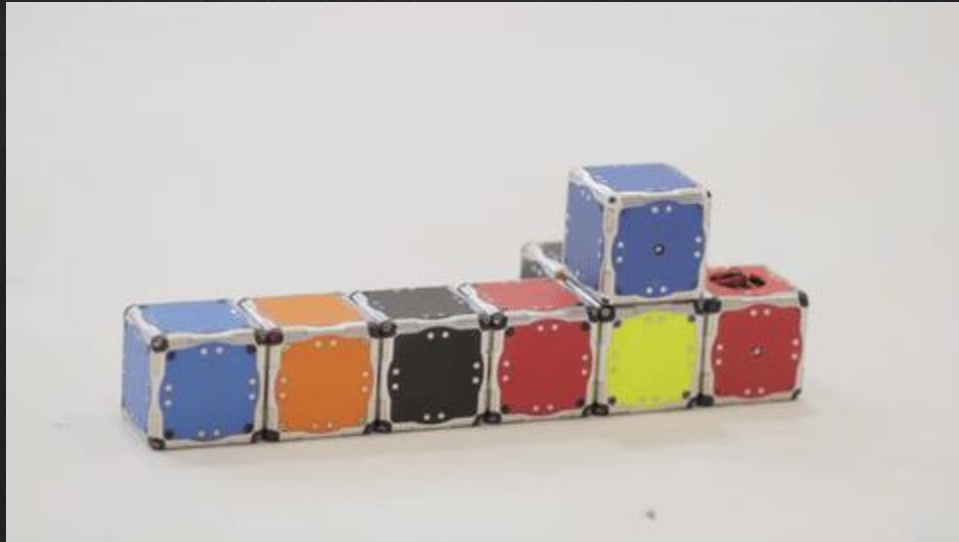
CROWDSOURCING: LET HUMANS FIND THE ANSWER



Playing video games for science:
You get smarter, we get data.

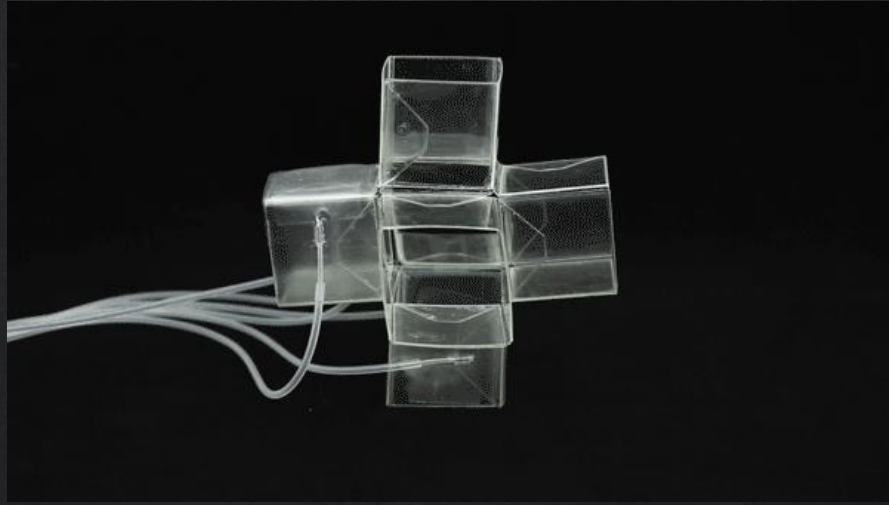
A mobile game app for finding
interlocking patterns designed by
people.

Having a heuristic would be great!



ROBOTIC USES

Polyominoes + Robots + Self-Assembly



SELF-INTERLOCKING

Foldings + Snapology = Interlocking 3D structures



THANKS!

Any questions?

You can find me at

@andresbeja87

abejara@purdue.edu

<https://www.cs.purdue.edu/homes/abejara/>

REFERENCES

All 18 pentominoes

(https://commons.wikimedia.org/wiki/File%3AAll_18_Pentominoes.svg)

By R. A. Nonenmacher (Created by me) [GFDL

(<http://www.gnu.org/copyleft/fdl.html>) or CC BY-SA 4.0-3.0-2.5-2.0-1.0

(<http://creativecommons.org/licenses/by-sa/4.0-3.0-2.5-2.0-1.0>)], via
Wikimedia Commons.