

6.849: Geometric Folding Algorithms

Prof. Erik Demaine

<http://courses.csail.mit.edu/6.849/fall10/>

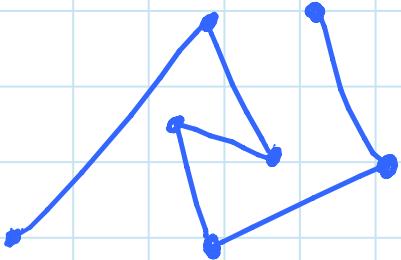
In general: Mathematics & algorithms behind
(un)folding of geometric objects

Applications/connections to:

- robotics → arms, Transformers, programmable matter, ...
- graphics → morphing, animation, ...
- mechanics → steam engines, ...
- manufacturing → sheet-metal & tube bending, nanomanufacturing, optics, ...
- medical → stents, drug delivery, ...
- aerospace → telescope deployment, ...
- biology → protein folding & design, ...
- sculpture → origami, interactive sculpture, ...
- architecture → dynamic architecture, deployable/collapsible structures, ...

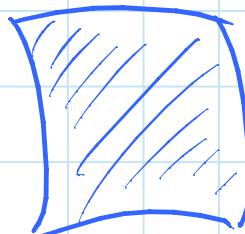
Geometric objects & rules for folding:

I linkage



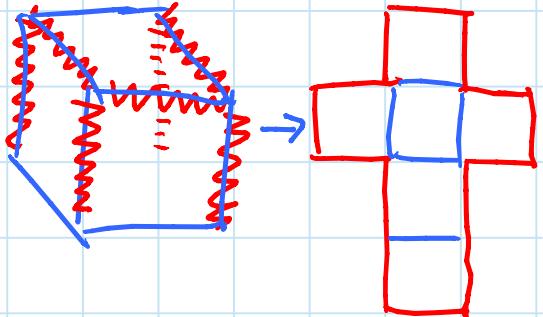
- ↪ rigid bars
- ↪ [don't cross]

II paper



- ↪ don't stretch
- ↪ don't tear
- ↪ don't cross

III polyhedron



- ↪ cut surface
- ↪ one piece
- ↪ no overlap

Questions:

- What structures can fold at all or in a particular way?
- What shapes (or other properties) can be folded, and how?

FOLDABILITY

DESIGN

Results:

- Everything is foldable!
(& here's an algorithm to do it)
- Efficient algorithm to decide foldability
- Computationally intractable
to decide foldability

UNIVERSALITY

DECISION

HARDNESS

The Class:

- lectures (mandatory attendance)
- problem sets (not a lot)
- project & presentation
 - implement algorithm/illustration/tool
 - sculpture/design
 - pose an open problem
 - Survey a subfield
 - try to solve an open problem
- open problem session (Optional)

I) LINKAGES: first, allowing intersection

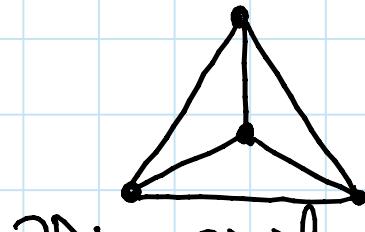
Early motivation: convert linear motion
(steam engines) \leftrightarrow circular motion

- Watt parallel motion [1784]
- Peaucellier inversor [1864]
- Kempe's How To Draw a Straight Line [1877]

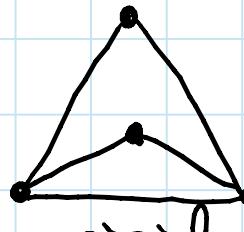
Universality: [Kempe 1876; Kapovich & Millson 2002;
King 1998; Abbott, Barton, Demaine, O'Rourke]

- there's a linkage signing your name
(tracing any polynomial curve) Erik
- **OPEN**: forbidding crossings?

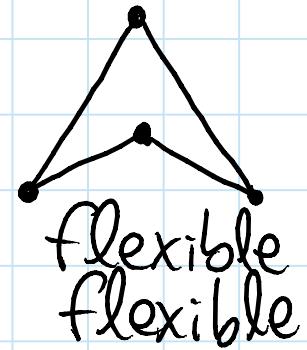
Rigidity: which linkages fold at all?



2D: rigid
3D: rigid



2D
flexible



2D
flexible

- efficient characterization in 2D
- **OPEN**: 3D

I) LINKAGES: forbidding intersection

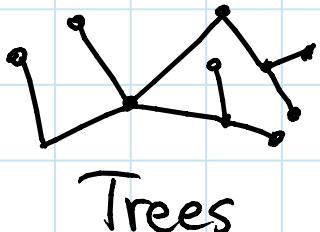
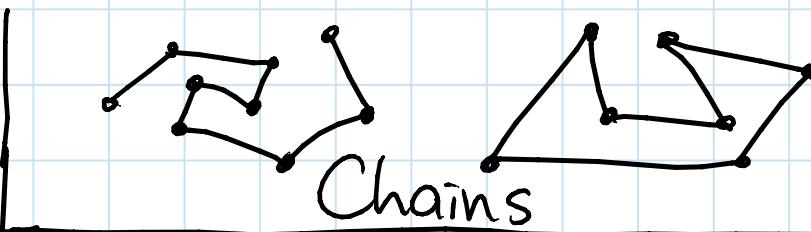
Reconfiguration: fold from config. A to config. B

- PSPACE-complete in general

polynomial space — likely exponential time

[Canny et al.; Alt, Knauer, Rote, Whitesides 2004]

- universality for special linkages:



2D	Always [Connelly, Demaine, Rote 2000]	Not [Biedl et al. 1998]
3D	Not [Cantarella & Johnston 1998]	Not
4D+	Always [Cocan & O'Rourke 2001] + algorithms	Always [ditto]

[Streinu 2000; Cantarella, Demaine, Iben, O'Brien 2004]

- **OPEN**: which 3D chains / 2D trees have locked configurations
(vs. can be folded between all configs.)
- whereas $A \rightarrow B$ problem is PSPACE-comp.
- protein folding leads to many cool problems

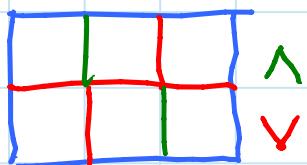
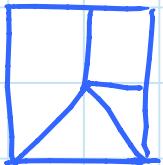
II PAPER:

Foldability: which crease patterns fold flat?

- NP-hard [Bern & Hayes 1996]

- efficient characterization for single vertex [Kawasaki; Justin; Hull 2003]

- **OPEN**: $2 \times n$ maps

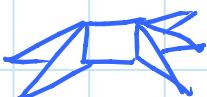


Design: what shapes can be folded?

- any 2D polygon, 3D polyhedron, 2-color pattern (inefficiently) [Demaine, Demaine, Mitchell 2001]

- practically: Origamizer [Tachi & Demaine]

- efficient "stick figures" for origami bases:



- TreeMaker [Lang 2003; Lang, Demaine, Demaine]

- NP-hard [Demaine, Fekete, Lang 2010]

- fold & cut: any set of line segments can be aligned by flat folding [Demaine, Demaine, Lubiw 1998; Bern, Demaine, Eppstein, Hayes 1998]

- curved creases [Huffman; Resch;

- Demaine, Demaine, Koschitz]

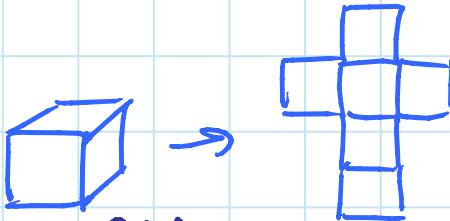
- universal "hinge patterns" (possible creases)

- [Bebennou, Demaine, Demaine, Ovadya 2010]

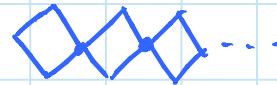
- self-folding sheets [Hawkes, Ah, Bebernou, Tanaka, Kim, Demaine, Rus, Wood - PNAS 2010]

III POLYHEDRA:

Unfolding:



- **OPEN**: edge-unfolding convex polyhedra [Dürer 1525]
- no conjectured counterexamples [Schlickenrieder 1997; Lucier 2006]
- every attempted algorithm fails [Bern et al. 2001]
- general unfolding (can cut anywhere)
- possible for convex [Agarwal et al. 1997; Sharir & Schorr 1986]
- possible for orthogonal [Damian, Flatland, O'Rourke 2006]
- **OPEN** in general
- vertex unfolding
- possible for triangulated [Demaine, Eppstein, Erickson, Hart, O'Rourke 2002]
- **OPEN** for convex [Erickson, Hart, O'Rourke 2002]



Folding: glue polygon boundary to make convex polyhed.

- exponential algorithm to list all gluings
(there can be that many) [Demaine, Demaine, Lubiw, O'Rourke 2002]
- poly. time for bounded sharpness [Lubiw, O'Rourke 2002]
- **OPEN**: poly. time decision / 1 shape?
- possible for "edge-to-edge" gluing [Lubiw & O'Rourke 1998]
- reconstructing resulting 3D shapes
[Bobenko & Izmestiev 2006;
Kane, Price, Demaine 2009]

IV HINGED DISSECTIONS

- any finite set of polygons of same area can be folded from one chain of polygons (without collision)

[Abbott, Abel, Charlton, Demaine, Demaine, Komninos
2008]