New Results in Folding

Erik Demaine
Martin Demaine
Jason Ku
Filling a Hole in a Crease Pattern:
Isometric Mapping from Prescribed Boundary Folding

Erik Demaine & Jason Ku

The 6th International Meeting on Origami in Science, Mathematics and Education

2014 August 11
Motivation
Problem

Input

\( \mathbb{R}^2 \)

\( \partial P \)

\( p \in V(P) \)

\( f(\partial P) \)

Output

\( \mathbb{R}^d \)

\( \mathbb{R}^2 \)

\( P \)

\( g(P) \)
Applications
Necessary Condition

**Proof.** Consider the following $d$-dimensional balls: $S_0$ centered at $f(q)$ with radius $k_p q$, $S_1$ centered at $f(u)$ with radius $k_p u$, and $S_2$ centered at $f(v)$ with radius $k_p v$ (See Figure 2). Because $\{q, u, v\}$ is critical and $\{p, u, v\}$ is nonexpansive under $f$, $f(p) \subseteq S_1 \setminus S_2 \implies S_0$. Because $f(p)$ is in $S_0$, $k_p q f(p) = k_p f(q)$ and $\{p, q\}$ is nonexpansive under $f$.

We will consider a polygon $P$ to be a closed figure in $\mathbb{R}^2$ bounded by three or finitely many more line segments that terminate in pairs at the same number of vertices, and do not intersect other than at their vertices (adapted from [2]). This definition restricts polygons to topological disks, and allows adjacent edges to be collinear. Let $V(P)$ denote the vertices of $P$, $\partial P$ denote the boundary of $P$, with $V(P) \implies \partial P \implies P$. An edge of $P$ is a line segment in $\partial P$ with endpoints at adjacent vertices. We will also say that a point $p \in P$ is visible from a vertex $v \in V(P)$ if the line segment from $p$ to $v$ is in $P$.

With the terminology in place, we can now clearly state the problem. Given a polygon $P$ in the plane with a mapping of its boundary $\partial P$ into $\mathbb{R}^d$, we desire an isometric mapping of $P$ into $\mathbb{R}^d$ (see Figure 3).

Because we would like to output a finite crease pattern, it only makes sense to look for mappings of $P$ into $\mathbb{R}^d$ for $d \geq 2$, so $d$ will be thus restricted for the remainder.

**Definition.** *(Valid Mapping)* Given polygon $P$ and mapping $f: \partial P \to \mathbb{R}^d$, define $f$ to be valid if $\partial P$ is nonexpansive under $f$ and adjacent vertices of $P$ are critical under $f$.

**Input Map:** Nonexpansive

![Diagram](Image)
FILLING A HOLE IN A CREASE PATTERN

Proof. Consider the following $d$-dimensional balls: $S_0$ centered at $f(q)$ with radius $k^p q$, $S_1$ centered at $f(u)$ with radius $k^p u$, and $S_2$ centered at $f(v)$ with radius $k^p v$ (See Figure 2). Because $\{q, u, v\}$ is critical and $\{p, u, v\}$ is nonexpansive under $f$, $f(p) \in S_1 \setminus S_2 \implies S_0$. Because $f(p)$ is in $S_0$, $k^p q \leq k^p f(p) \leq k^p f(q)$ and $\{p, q\}$ is nonexpansive under $f$.

We will consider a polygon $P$ to be a closed figure in $\mathbb{R}^2$ bounded by three or finitely many more line segments that terminate in pairs at the same number of vertices, and do not intersect other than at their vertices (adapted from [2]). This definition restricts polygons to topological disks, and allows adjacent edges to be collinear. Let $V(P)$ denote the vertices of $P$, $\partial P$ denote the boundary of $P$, with $V(P) \implies \partial P \implies P$. An edge of $P$ is a line segment in $\partial P$ with endpoints at adjacent vertices. We will also say that a point $p \in P$ is visible from a vertex $v \in V(P)$ if the line segment from $p$ to $v$ is in $P$.

With the terminology in place, we can now clearly state the problem. Given a polygon $P$ in the plane with a mapping of its boundary $\partial P$ into $\mathbb{R}^d$, we desire an isometric mapping of $P$ into $\mathbb{R}^d$ (see Figure 3). Because we would like to output a finite crease pattern, it only makes sense to look for mappings of $P$ into $\mathbb{R}^d$ for $d \geq 2$, so $d$ will be thus restricted for the remainder.

Definition. (Valid Mapping) Given polygon $P$ and mapping $f: \partial P \to \mathbb{R}^d$, define $f$ to be valid if $\partial P$ is nonexpansive under $f$ and adjacent vertices of $P$ are critical under $f$.

Input Map: Nonexpansive & Sufficient
**Local Solution**

\( \mathbb{R}^2 \)

- Points: \( u, v, w \)
- Angle: \( \theta \)
- Region: \( R \)
- Set: \( S \)
- Angle: \( \beta \)

\( \mathbb{R}^d \)

- Points: \( f(u), f(v), f(w), q \)
- Angle: \( \phi \)
- Region: \( f(R) \)
Split Point
Software: 2D
3D Application
Algorithmic Lower Bounds: Fun with Hardness Proofs

Super Mario Bros.

Crossover gadget for NP-hardness

Rush Hour

AND gadget for PSPACE-hardness

Minesweeper

OR gadget for NP-hardness

Hardness Made Easy*

Learn when to give up the search for efficient algorithms; see connections between computational problems; solve puzzles to prove theorems, solve open problems, and write papers.

Topics: NP, PSPACE, EXPTIME, EXPTIME, EXPSPACE, 3SUM, approximation, fixed parameter, games & puzzles, 3SUM, key problems, gadgets, and proof styles.

6.890 taught by Professor Erik Demaine

Fall 2014
Examples of hardness proofs:

1. *Super Mario Bros.* is NP-complete
   - reduction from 3SAT: can you satisfy (make true) a formula like
     \[(x_1 \text{ or } x_2 \text{ or } x_3) \text{ AND } (x_5 \text{ or } \neg x_3 \text{ or } x_4) \text{ and } \ldots\]

2. *Rush Hour* is PSPACE-complete
   - reduction from NCL (Nondet Constraint Logic): given directed graph with edge weights ∈ \{1, 2\}, find sequence of edge reversals to reverse a target edge, while at all times maintaining total in-weight ≥ 2 at each vertex
   - only need AND vertex & OR vertex

(in fact: OR can be protected: only one input active at once)

Constraint Logic is a powerful tool for proving hardness of games & puzzles

http://courses.csail.mit.edu/6.890/fall14/lectures/
Simple Folding is Strongly NP-Complete

6.890 Algorithmic Lower Bounds Final Project

Submitted to WADS 2015

Hugo A. Akitaya
Tufts University
hugoakitaya@gmail.com

Erik D. Demaine
MIT CSAIL
edemaine@mit.edu

Jason S. Ku
MIT FIL
jasonku@mit.edu
Simple Folds
Simple Folds
## Simple Fold Models

<table>
<thead>
<tr>
<th>Simple Folding Model</th>
<th>Restriction on # for ( q \in L )</th>
<th>Foldable Example Steps</th>
</tr>
</thead>
<tbody>
<tr>
<td>Some-layers</td>
<td>no restriction</td>
<td>(1), (2), (3), (4), (5), (6)</td>
</tr>
<tr>
<td>One-layer</td>
<td>( #(q) \in {0, 1} )</td>
<td>(1), (5)</td>
</tr>
<tr>
<td>All-layers</td>
<td>( #(q) \in {0, #_+(q)} )</td>
<td>(1), (2), (3)</td>
</tr>
<tr>
<td>Infinite Some-layer</td>
<td>( #(q) \geq 1 )</td>
<td>(1), (3), (4)</td>
</tr>
<tr>
<td>Infinite One-layer</td>
<td>( #(q) = 1 )</td>
<td>(1)</td>
</tr>
<tr>
<td>Infinite All-layers</td>
<td>( #(q) = #_+(q) )</td>
<td>(1), (3)</td>
</tr>
</tbody>
</table>

![Foldable Example Steps](image-url)
Computational Complexity

<table>
<thead>
<tr>
<th>Model</th>
<th>Assigned</th>
<th>Unassigned</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>open</td>
<td>poly</td>
</tr>
<tr>
<td>One-layer</td>
<td>weak → strong</td>
<td>open</td>
</tr>
<tr>
<td>Some-layers</td>
<td>weak → strong</td>
<td>weak → strong</td>
</tr>
<tr>
<td>All-layers</td>
<td>weak → strong</td>
<td>weak → strong</td>
</tr>
<tr>
<td>Inf. One-layer</td>
<td>strong</td>
<td>open</td>
</tr>
<tr>
<td>Inf. Some-layers</td>
<td>strong</td>
<td>open</td>
</tr>
<tr>
<td>Inf. All-layers</td>
<td>open</td>
<td>open</td>
</tr>
</tbody>
</table>

**weak** = weakly NP-complete  
**strong** = strongly NP-complete  
**poly** = polynomial-time algorithm  
**open** = unsolved problem
3-Partition

• Given \( n \) integers \( a_1, a_2, \ldots, a_n \), can you partition into \( n/3 \) triples with the same sum?
  
  • Know the sum must be \( T = \frac{\sum i \cdot a_i}{n/3} \)

• This problem is strongly \textbf{NP-complete}: NP-complete even if \( a_i \) numbers are \( n^{O(1)} \)

\[
\begin{align*}
\text{?} & \quad \rightarrow \\
T & \quad \downarrow \\
\ldots & \quad \downarrow \\
n/3 \text{ triples} & \quad \downarrow
\end{align*}
\]
New: Strongly NP-Hard

Fig. 3. Process to check the Partition solution: 1) pleat variables to change height of bar by $2t$, 2) fold along the rightmost wrapper crease around the column, 3) fit the bar through the cage folding the bar to the left along the next wrapper crease, 4) repeat until $n/3$ triples adding to $2t$ have been checked.

Proof Sketch: The reduction is from $3$-Partition by constructing the orthogonal polygon shown in Figure 2 from a $3$-Partition instance, with $n$ integers to be partitioned into $n/3$ triples that each sum to $t$, that is simply-foldable if and only if the instance is a "yes" instance. A broad picture of the construction is shown in Figure 3. The integers of the $3$-Partition are encoded in the Staircase. The Bar is a long uncreased section of the paper that must fold through the Cage $n/3$ times, enforced by creases in the Wrapper section. The Bar can only pass through the Cage each time if it raises by exactly $2t$ from the last time it passed through the Cage, which can only occur if creases have been folded from a triple of integers summing to $t$. Passing through the Cage $n/3$ times checks the $3$-Partition instance; otherwise, the construction is not simply-foldable.

The optimization version of the decision problem is even hard to approximate. Theorem 2. Given an orthogonal paper with paper-aligned orthogonal creases admitting a maximum sequence of $m$ simple folds, approximating MaxFold to within a factor of $m^{1/2}$ for any constant $\epsilon > 0$ is NP-complete in the some-layers and all-layers models.

Proof Sketch: In the above reduction, adding a bunch of horizontal creases in the Wrapper section intersecting the vertical creases, appropriately M/V assigned, constrain them not to be simply-foldable unless all the creases in the above reduction have folded. Adding enough horizontal creases achieves the claimed bound.
New: Strongly NP-Hard

Reduction from 3-Partition

Given integers:
\[ A = \{a_1, \ldots, a_n\} \]

Find partition:
\[ A = \bigcup_{i=1}^{n/3} A_i \]
\[ \sum_{A} \frac{A}{n/3} = \sum A_i = t \]
New: Strongly NP-Hard

Reduction from
3-Partition

Given integers:
\[ A = \{a_1, \ldots, a_n\} \]

Find partition:
\[ A = \bigcup_{i=1}^{n/3} A_i \]
\[ \sum_{A} = \sum A_i = t \]
New: Strongly NP-Hard

Reduction from 3-Partition

Given integers:
\[ A = \{a_1, \ldots, a_n\} \]

Find partition:
\[ A = \bigcup_{i=1}^{n/3} A_i \]
\[ \sum_{n/3}^A A = \sum A_i = t \]
New: Strongly NP-Hard

Reduction from
3-Partition

Given integers:
\[ A = \{a_1, \ldots, a_n \} \]

Find partition:
\[ A = \bigcup_{i=1}^{n/3} A_i \]
\[ \sum_{i=1}^{n/3} A_i = \sum A_i = t \]
New: Strongly NP-Hard

Reduction from 3-Partition

Given integers:
\[ A = \{a_1, \ldots, a_n\} \]

Find partition:
\[ A = \bigcup_{i=1}^{n/3} A_i \]
\[ \frac{\sum A}{n/3} = \sum A_i = t \]
Square: Assigned

Arkin et al. adapt their Partition reduction to square paper with M/V assigned paper-aligned creases at multiples of 45 by constructing an approximation of their orthogonal construction from a square. Unfortunately their modification cannot be applied to our 3-Partition reduction in the all-layers model because their construction requires folds along the long construction end which may (will) overlap other parts of the paper during construction.

Instead, we use a similar idea to construct an orthogonal polygon approximation from a square but with a different turn gadget that enforces the order of construction while only making folds local to the gadget that works in both the some-layers and all-layers models.

Theorem 3. The assigned simple-foldability decision problem for square paper with paper-aligned creases at multiples of 45 is strongly NP-complete in the some-layers and all-layers models.

Proof Sketch: The reduction is an extension of the reduction used in the proof of Theorem 1. We construct an approximation of the orthogonal polygon in Figure 2 from a square by creating a long, thin rectangle which we then manipulate into the desired shape by using the turn gadgets shown in Figure 4. When chained together in a sequence, the turn gadgets have the property that one cannot be folded until all turn gadget before it have been completed. We use this chained dependence to propagate a "signal" that preceding creases have folded completely. By starting the construction of the orthogonal polygon from the Cage, the entire orthogonal polygon must be folded before any Wrapper crease can fold, at which point Theorem 1 applies.

Square: Unassigned

M/V unassigned crease patterns are naturally less restrictive than M/V assigned crease patterns. This freedom can make collision avoidance easier, providing a choice folding direction at each crease. However when proving hardness for M/V unassigned crease patterns, one cannot use crease direction to enforce fold ordering or layering and must restrict them using other techniques. Arkin et al. provide a weakly NP-hard reduction for orthogonal polygons with unconstrained creases without crease assignment in Theorem 7.1, but their proof has two errors. We discuss these errors in the Appendix but focus here on showing stronger models strongly NP-complete.

Theorem 6. The assigned simple-foldability decision problem for orthogonal paper with paper-aligned orthogonal creases is strongly NP-complete in the infinite one-layer and infinite some-layers models.

Proof Sketch: The proof is again a reduction from 3-Partition via the orthogonal crease pattern shown in Figure 8. There are four sets of creases: horizontal creases \{h_1, h_2\}, variable creases \{v_i\}_{i=1}^{V}, set creases \{s_i\}_{i=1}^{S}, and checker creases \{c_i\}_{i=1}^{C}. The horizontal creases must fold before creases in \{C \cup S\} or \{V\} can fold.
In the infinite one-layer or some-layers models, a simple fold must fold one (or more) layer(s) everywhere in the intersection of the fold axis and the valid flat folding. This is more restrictive than the one-layer model as foldability in the infinite one-layer model implies foldability in the one-layer model but not the reverse.

Theorem 6. The assigned simple-foldability decision problem for orthogonal paper with paper-aligned orthogonal creases is strongly NP-complete in the infinite one-layer and infinite some-layers models.

Proof Sketch: The proof is again a reduction from 3-Partition via the orthogonal crease pattern shown in Figure 8. There are four sets of creases: horizontal creases \( \{h_1, h_2\} \), variable creases \( v_i \), set creases \( s_i \), and checker creases \( c_i \). The horizontal creases must fold before creases in \( C \) or \( V \) can fold,
Flat Folding with Thick Materials

ASME IDET/CIE 2015

Erik D. Demaine
MIT CSAIL

Jason S. Ku
MIT FIL
Thick Folding Approaches

A

B

C

D
New Approach
Layer Ordering Graph
Offset Layers
Shrink Polygons
Construct Crease Width
Polygon Crossings
Self-Intersections
Simulation
Model