<table>
<thead>
<tr>
<th>タイトル</th>
<th>じゃばら折りの一般化とその複雑さの研究</th>
</tr>
</thead>
<tbody>
<tr>
<td>作者</td>
<td>上原 隆平</td>
</tr>
<tr>
<td>引用</td>
<td>2010年度科学技術振興機構 ERATO湊離散構造処理系プロジェクト講究録 p.67-73.</td>
</tr>
<tr>
<td>発行日</td>
<td>2011-06</td>
</tr>
<tr>
<td>DOI</td>
<td></td>
</tr>
<tr>
<td>リンク</td>
<td><a href="http://hdl.handle.net/2115/48475">http://hdl.handle.net/2115/48475</a></td>
</tr>
<tr>
<td>タイプ</td>
<td>conference presentation</td>
</tr>
<tr>
<td>追加情報</td>
<td>10_all.pdf</td>
</tr>
</tbody>
</table>

HOKKAIDO UNIVERSITY
ERATO セミナ 2010 - No. 10
じゃばら折りの一般化とその複雑さの研究
上原隆平
北陸先端科学技術大学院大学 准教授
2010/8/4

概要
日本での折紙の研究は、数学的な観点からの研究が先行してきた。しかし Lang, Domain, O’Rourke といった研究者の活躍により、最近 computational origami という言葉が市民権を得つつある。しかし理論計算機科学としての枠組みは、まだまだ未整備である。折紙を計算機科学の対象としたときの単純なモデルとは何だろうか。ここではごく単純な折紙モデルとして、次のものを考える。入力として与えられるのは、長さ n+1 の紙と、長さ n の M, V 上の文字列である。ここで M は山折り、V は谷折りを意味している。つまり長さ n+1 の紙の上に、等間隔に山/谷が指定されている。この文字列にしたがって紙を折る問題を考える。まず時間計算量に対しては、折りの回数が自然に対応すると考えることができる。この観点から、じゃばら折りとその一般化に対して、講演者らは、ほぼ最適な折りアルゴリズムを設計した。では領域計算量に対応する概念は何だろうか。これに対してはみんなが合意する概念は存在しない。講演者は、折り目を挟まる紙の枚数を基準として提案し、これに関する最適化問題を研究している。部分的な解は得られているものの、まだ残された研究課題の方が多いのが実状である。
Background story...

- Toshikazu Kawasaki, a mathematician and designer of Kawasaki rose, said that "For a mathematician, it is OK if solution exists."

- A computer scientist, or I, cannot agree; "For a mathematician, it is OK if solution exists."

- How to find the solution and the cost to find/construct the solution.

- Goodness good algorithm

- Hardness computational complexity

- Is there any good problem just about computational cost??

• Pleat folding = 1D Origami

- Alternating foldings of Mountain and Valley
- Basic tool of Origami
- Many applications
- Extension to General Patterns and consider its complexity...

http://www.jaist.ac.jp/~uehara/etc/origami/

Complexity of folding(?)

From the viewpoint of Computer Science
- Two resources of an Origami model?
  1. time...the number of foldings (operations)
  2. space...minimization of "stretch" that is the number of papers between two hinged papers

http://www.jaist.ac.jp/~uehara/etc/origami/

JAIST
uehara@jaist.ac.jp
http://www.jaist.ac.jp/~uehara
or, ask Google with "uehara origami"
New open problem

- **Least stretch** folding problem
  
  **Input:** Paper of length $n+1$ and $s \in \{M, V\}^*$
  
  **Output:** folded paper according to $s$
  
  **Goal:** Find a **good** folded state with few **stretch**
  
  - At each crease, the number of papers between the papers hinged at the crease is **stretch**.
  
  - Two minimization problems; max/total (average)
    - minimize maximum
    - minimize total (average)

  
  It seems simple, so easy??

  Not!!

Least stretch folding problem

- **Partial answers:**
  
  1. The number of folding ways for a **random** pattern
    
    - $\Theta(1.65^n)$ by experiments
    
    - $\Omega(1.53^n)$ and $\Omega(2^n)$ by theoretical lower/upper bounds
      
      - so a naive program runs veeeeerrrrrry slow.

  
  **Note**
  
  These results are based on enumeration & rough counting, described hereafter...

Least stretch folding problem

- **Simple non-trivial example**
  
  **Input:** MMV/MMV/MMV
  
  The number of feasible folded states, **100**
  
  **Goal:** Find a **good** folded state with few **stretch**
  
  - The unique solution having min. max. value 3
    
    [5|4|3|6|7|2|8|10|12|11|9]
    
    total=13

  - The unique solution having min. total value 11
    
    [5|4|3|1|2|6|7|8|10|12|11|9]
    
    total=13

Least stretch folding problem

- **Open problem:**
  
  - **Tractable/Intractable?**
  
  - Poly-time solvable by Dynamic Programming?
    
    - up to now, I have no answer to this question ;-)!

  - Then, exhaust search technique, that produces all possible folding ways, works?

  exponential pattern
  
  average? pleats

  Not!!

Least stretch folding problem

- **1. Average number of folding ways for a random pattern $= f(n)/2^n$, where**
  
  $f(n) = \#$ of folding ways (or folded states) of a paper of length $n+1$ (summation for all possible patterns)

  - I give some bounds of $f(n)$:
    
    - "The On-Line Encyclopedia of Integer Sequences" tells us up to $n=28$ (by enumeration);
      
      $f(n) \sim 6(3.3^n)$.

    - I have upper/lower bounds;
      
      - upper bound: $f(n)=O(4^n)$
      
      - lower bound: $f(n)=\Omega(3.07^n)$
Least stretch folding problem

- Upper bound $f(n) = O(4^n)$ comes from the Catalan number.

[Proof] If the paper of length $n+1$ is folded, the crease points should be nested.

\[ \begin{array}{c}
\text{Combination of } w_2 \text{ pairs of } i = \text{Catalan Number } C_{n+2} \\
\end{array} \]

Least stretch folding problem

- Lower bound $f(n) = \Omega(3.07^n)$.

[Proof] We consider “folding of the last $k+1$ unit papers”;

\[ f(n) \geq (g(k))^2 = (g(k)^{k+1})^2 \]

From that site, we have $g(43) = 830776205506531894760$.
Thus, by $f(n) \geq (g(k))^2 = (g(k)^{k+1})^2$
we have the lower bound.

Open problem and Future work

- Ext. of Least stretch folding problem:
  - non-unit length
    - First of all, there is a pattern that cannot be folded;
    - revisit unit length paper...
  - Tractable/Intractable?
    - $\text{NP}$-hard?
    - Poly-time solvable by Dynamic Programming?
    - Fixed parameter tractable for “stretch $k$”?
  - What is the most complex pattern of M/V that has the most feasible folded states?
  - Extension to...
    - non-unit length (operation-restricted linkage)
    - 2D (a kind of map folding)
  - What ”space complexity” of Origami is?
    - area for folding?

Universal theorem:

- Universal theorem of unit length folding in a simple folding model;
  - Input: a folded state of a paper strip of length $n+1$ to unit length
  - Question: Is it foldable on a suitable set of operations?
  - Simple folding model

1. (from flat state)
2. pick up a point
3. valley fold most inner layers
4. to flat state
Universal theorem of linkage…

- Universal theorem of unit length folding in a simple folding model;
  - Any flat folded state of unit length can be made by a sequence of simple foldings of length at most 2n.

- Related results;
  - Any M/V pattern can be flat folded by repeating "end-folding"
  - If the creases are not unit-length, some folded state cannot be folded by simple foldings.
  - If "any folding" is allowed, "Yes" even in non-unit length
    - comes from "no locked linkage in 2D"!

[Phase 1] unveil the covered endpoint
- while we cannot see the endpoint, unfold the paper to expose the covered paper close to the endpoint.
- after at most n unfoldings, we can see the endpoint.

[Phase 2] peel the paper from the endpoint
- unfold and extend the last flat part that contains the endpoint.
- after at most n unfoldings, we obtain the flat paper.

[Note] At Phase 1, some new creases can be made by an "unfolding". Hence the total number of "unfolding" cannot be bounded by n.

Universal theorem of linkage…

Pleat folding

- Pleat folding (in 1D)
  - Repeating of mountain and valley foldings
  - Basic operation in some origami
  - Many applications

Repeating "folding in half" is the best way to make many creases

- Naïve algorithm: n time folding is a trivial solution
- We have to fold at least \(\log n\) times to make \(n\) creases
- More efficient ways…?
- General Mountain/Valley pattern?

- proposed at Open Problem Session on CCCG 2008 by R. Uehara.
Complexity of Pleat Folding

[Main Motivation] Do we have to make \( n \) foldings to make a pleat folding with \( n \) creases??

1. The answer is "No"!
   - Any pattern can be made by \( [n/2]+[n \log n] \) foldings
2. Can we make a pleat folding in \( o(n) \) foldings?
   - Yes!! It can be folded in \( O(\log^2 n) \) foldings.
3. Lower bound; \( \log n \)
   - \( \Omega(\log n \log \log n) \) lower bound for pleat folding!!

[Next Motivation] What about general pleat folding problem for a given M/V pattern of length \( n \)?

- Any pattern can be made by \( [n/2]+[n \log n] \) foldings
  1. Upper bound:
     - Any M/V pattern can be folded by \( (4+\epsilon) \frac{n}{\log n} + \left(\frac{n}{\log n}\right) \) foldings
  2. Lower bound:
     - Almost all mountain/valley patterns require \( \frac{n}{3+\kappa n} \) foldings

[Note] Ordinary pleat folding is exceptionally easy pattern!

Upper bound of Unit FP (1)

- Any pattern can be made by \( [n/2]+[n \log n] \) foldings
  1. \( M/V \) fold at center point according to the assignment
  2. Check the center point of the folded paper, and count the number of 1's and 0's (we have to take care that odd depth papers are reversed)
  3. \( M/V \) fold at center point taking majority
  4. Repeat steps 2 and 3
  5. Unfold all (cf. on any model)
  6. Fix all incorrect crease points one by one

Steps 1\~4 require \( \log n \) and step 6 requires \( n/2 \) foldings

Mountain folding in \( \log^2 n \) foldings

Step 1:
1. Fold in half until it becomes of length \( [vvv] \) (log n-2 foldings)
2. Mountain fold 3 times and obtain \([MMM]\)
3. Unfold; \( vMMMvMvMMvMvM\ldots\)

Step 2:
1. Fold in half until all "v"'s are piled up (log n-3 foldings)
2. Mountain fold 5 times \([MMMvMM]\), and unfold
3. \( vMMMvMMvMMvMMvMMvMMvMMvMMvMM\ldots\)

Step 3; Repeat step 2 until just one "v" remains
\( vMMMvMMvMMvMMvMMvMMvMMvMMvMMvMMvMMvMM\ldots\)

Step 4; Mountain fold all irregular vs step by step.

- \#iterations of Steps 2\~3; \( \log n \)
- \#valleys at step 4; \( \log n \)
- \#foldings in total= (log n)
• Lower bound of Unit FP

[Thm] Almost all patterns but o(2^n) exceptions require \( \Omega(n/\log n) \) foldings.

[Proof] A simple counting argument:

1. # patterns with \( n \) creases > 2^{n/2} = 2^{n-2}
2. # patterns after \( k \) foldings < (2 \times n) \times (n+1) \times (2 \times n) \times \ldots \times (n+1) \times (2 \times n)

We cannot fold most patterns after at most \( k \) foldings if

\[
\sum_{i=0}^{k} (2n(n+1))^i \leq (2n(n+1)+1)^k < 2^{n/2}
\]

Letting

\[
n \geq 2, k = O\left(\frac{n}{\log n}\right)\text{ we have } (2n(n+1)+1)^k = o(2^n)
\]

Any pattern can be folded in \( cn/\log n \) foldings

Prelim.

- Split into chunks of size \( b \):
  1. Each chunk is small and easy to fold
  2. # kinds of different \( b \) are not so big

Main alg.

- For each possible \( b \)
  1. pile the chunks of pattern \( b \) and mountain fold them
  2. fix the reverse chunks
  3. fix the boundaries

Repeat for all chunks

Analysis is omitted

Open Problems

- Pleat foldings
  - Make upper bound \( O(\log^2 n) \) and lower bound \( \Omega(\log n / \log \log n) \) closer

- “Almost all patterns are difficult”, but…
  - No explicit M/V pattern that requires \( (cn/\log n) \) foldings

- When “unfolding cost” is counted in…
  - Minimize #foldings + # unfoldings