実践的幾何アルゴリズム
Advanced Algorithms for Computational Geometry

14. 最新のアルゴリズムの話題から：計算折り紙その2

担当：上原隆平

忘れないようにアナウンス：

- 11月28日（火）：最後の講義.
  - 10:30- 講義アンケートと，試験のリクエスト
  - オフィスアワーは補講はしません.
- 11月30日（木）：期末試験

Advanced Algorithms for Computational Geometry
実践的幾何アルゴリズム

14. Recent Topics in Advanced Algorithm: Computational Origami II

Ryuhei Uehara

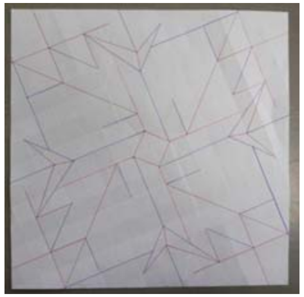
今日のトピック

「折り」の**アルゴリズムと計算量**の関係

- 折り紙の基本操作
- 折り紙のアルゴリズムと計算量
  - １次元の紙における効率のよい折り方（アルゴリズムと計算量）
    - 高速に折るアルゴリズム（折る回数を減らせるか？）
    - 「良い折り畳み状態」を評価する指標のモデル
  - １次元の紙における計算不能性（計算の理論）
    - 計算モデル

未解決問題が多く、若手の活躍がめざましい分野です．

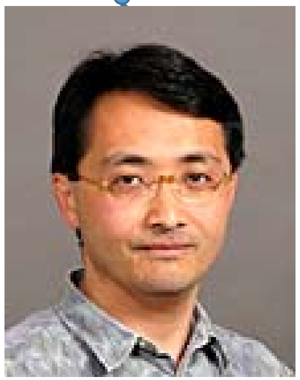
今日の話の背景



川崎ローズ

- 2008年6月22日、
  第4回「折り紙の科学・数学・教育研究会」にて、、、
  - 川崎敏和氏(数学者だけど川崎ローズの作者として有名)いわく:
    「数学者としては、解の存在さえわかれば、あとはどうでもいい」

どうでもいい余談
九州大学の川崎英文先生
(ORの研究者)とは双子です.



- 計算機科学者である上原は…
  - 解の求め方とその
    計算コストが大切!!
    - 良いアルゴリズム
    - 計算量的な困難さ

良い問題は
ないかなぁ

折り紙の「計算量」の最終ゴール

「どちらが複雑か？」という問いに答えを与えたい！

川崎ローズ

3次元空間で折る必要がある。
上原は最初10日間かかったが、
今では10分くらいで折れる。

テキストを見ながら折ると、
上原は40分くらいで折れる。
前川さんは20分くらいで折れる。

前川の悪魔

どちらも少なくとも折り鶴よりは「複雑」だと思うけれど、
それはなぜか？どういう意味で複雑なのか？

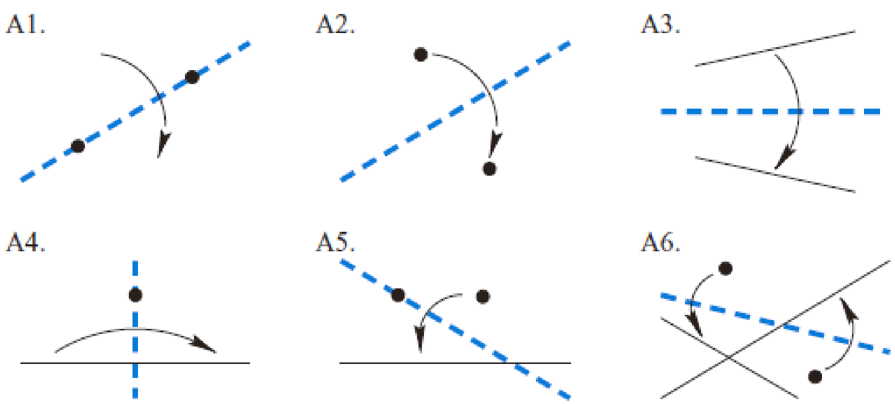
計算量の理論とアルゴリズム理論

- 理論計算機科学の基礎理論
  - 基準となるマシンモデル：
    共通の「<u>基本演算</u>」に関する合意が必要
    - チューリングマシン
    - VRAMモデル
  - <u>アルゴリズム</u>
    - 基本演算をどのような手順で組み合わせるか？
  - アルゴリズムの<u>計算量</u>
    - 時間計算量：基本演算の回数で効率を測る
    - 領域計算量：計算に必要な記憶領域で効率を測る

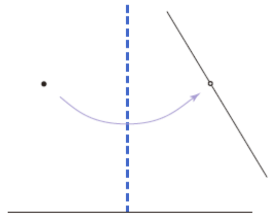
折りの「複雑さ」…？

この7操作は4次方程式まで解ける。
（定規とコンパスでは2次方程式までしか解けない）

- 折り紙のモデル:
  - 折り紙業界では「7種の操作」が合意されている。
  - これを「基本演算」と考えてよさそう。



フジタの公準A1-A6



羽鳥の操作A7

折りの「複雑さ」…？

- 折り紙の計算量的な複雑さを考えるにあたって，
  妥当なモデルとは？（単純→複雑）
  1. 最も単純なモデル：1次元で等間隔な折り紙
     - 長い紙テープ上に，等間隔に折り目をつける/与えられる
  2. 拡張の方向は二つ
     - 折り目が等間隔でなくてもよい
       - （斜めも許す？）
     - ２次元や高次元への拡張

今は，まだ
ほぼこのあたり！

折り紙の複雑さ/効率(?)

- 計算機科学の視点で考えよう…
- チューリング機械モデルにおける2種類の<u>**資源**</u>
  1. <u>時間</u>：基本演算の適用回数
  2. <u>領域</u>：計算に必要なメモリ量

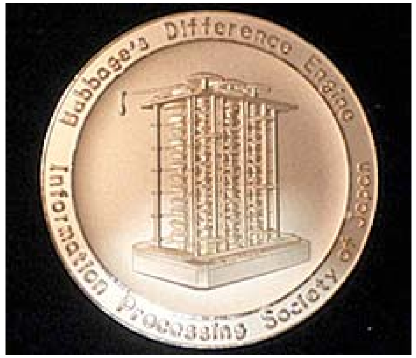
折り紙の複雑さ/効率(?)

- 計算機科学の視点で考えよう…
- 折り紙モデルにおける2種類の資源とは？
  1. 時間…折り(基本演算)の回数
     - J. Cardinal, E. D. Demaine, M. L. Demaine, S. Imahori, T. Ito, M. Kiyomi, S. Langerman, R. Uehara, and T. Uno: Algorithmic Folding Complexity, *Graphs and Combinatorics*, Vol. 27, pp. 341-351, 2011.
  2. 領域…???
     - R. Uehara: Stretch Minimization Problem of a Strip Paper, *5th International Conference on Origami in Science, Mathematics and Education*, 2010/7/13-17.
     - R. Uehara: On Stretch Minimization Problem on Unit Strip Paper, *22nd Canadian Conference on Computational Geometry*, pp. 223-226, 2010/8/9-11.

今日の話題

5. 時間計算量
   - “Folding complexity”入門
     - 理論上、世界最速のジャバラ折りアルゴリズム
6. 領域計算量（？）
   - 切手折り問題
   - 折り目幅最小化問題
     - NP完全問題、FPTアルゴリズムなど
7. （折り紙における決定不能問題）
   - 対角線論法と不完全性定理

The 7th EATCS/LA Presentation Award!



2012年3月
情報処理学会
山下記念研究賞

その一方：ある論文誌では「証明が簡単すぎる」という理由で reject されました ;-)
新たなモデルの提案はむつかしいです。。。

演習問題(研究課題)

- 演習問題
  - 折り紙の「複雑さ」を評価するための指標を提案せよ。

    例:作業スペース
  - 上記の指標を吟味せよ。

    例:1次元の折り紙(パイプを曲げるなど)では意味があるが、2次元の正方形だと、だんだん小さくなる一方なので、あまりよくない。

野望:チューリング機械モデルにおける「time-space trade off」に相当するような複雑さの指標を提案したい!

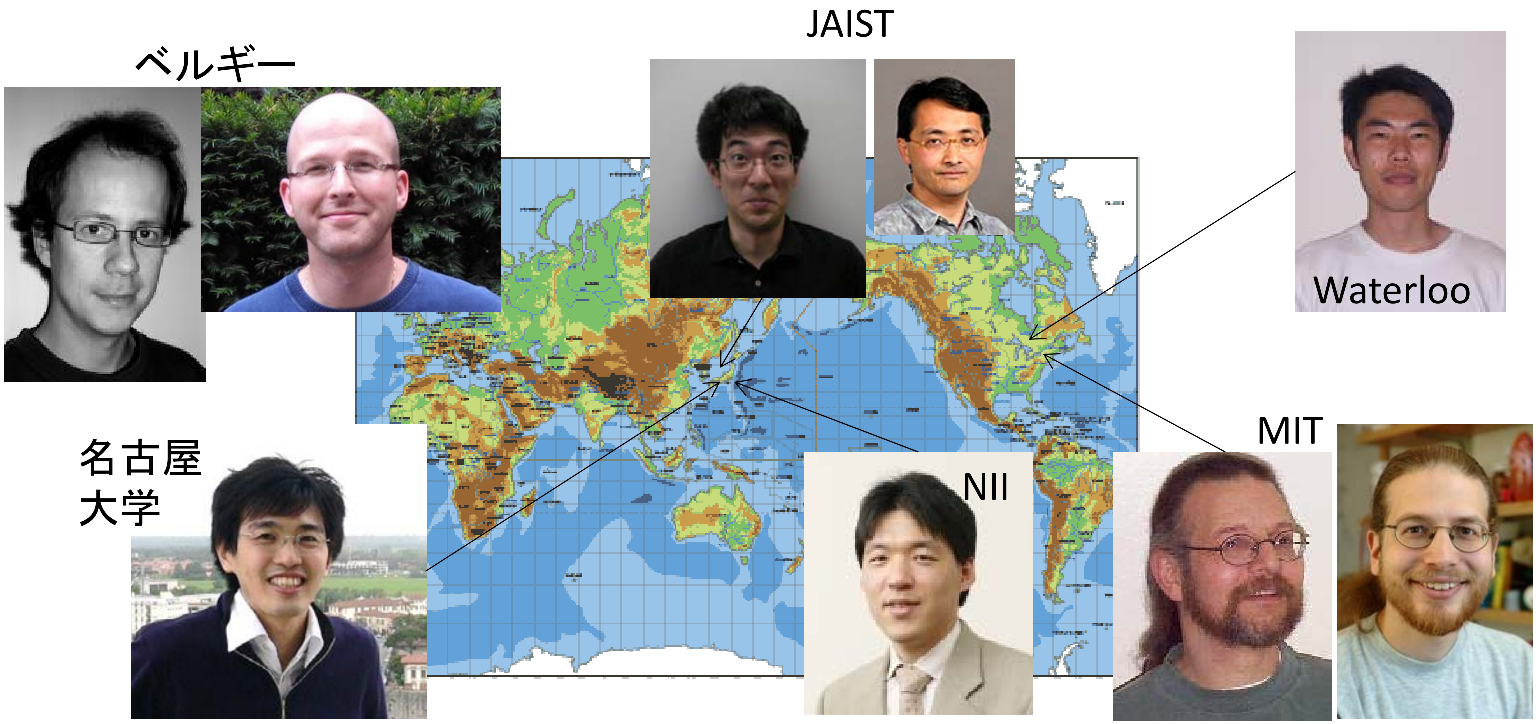
じゃばらを高速に折る
--- folding complexity ---

上原隆平（うえはらりゅうへい）
北陸先端科学技術大学院大学　情報科学研究科
Japan Advanced Institute of Science and Technology (JAIST)
uehara@jaist.ac.jp
http://www.jaist.ac.jp/~uehara

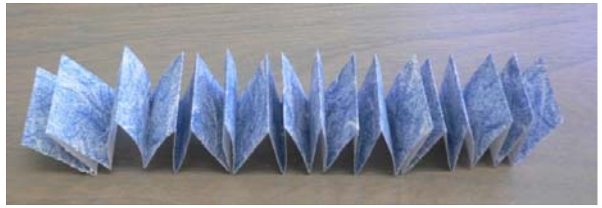
主要な論文
J. Cardinal, E. D. Demaine, M. L. Demaine, S. Imahori, T. Ito, M. Kiyomi, S. Langerman, R. Uehara, and T. Uno: Algorithmic Folding Complexity, *Graphs and Combinatorics*, Vol. 27, pp. 341-351, 2011.

1. 時間…折り（基本演算）の回数

- J. Cardinal, E. D. Demaine, M. L. Demaine, S. Imahori, T. Ito, M. Kiyomi, S. Langerman, R. Uehara, and T. Uno: Algorithmic Folding Complexity, *Graphs and Combinatorics*, Vol. 27, pp. 341-351, 2011.



じゃばら折り = 1次元折り紙



- 山折りと谷折りが等間隔に交互に続く
- 折り紙の基本ツール
- 応用も…?

➢山/谷を一般化したときの複雑さとは?

「(M/Vによる)2進数文字列上の操作の問題」と考えると、コンピュータサイエンスと相性がよい

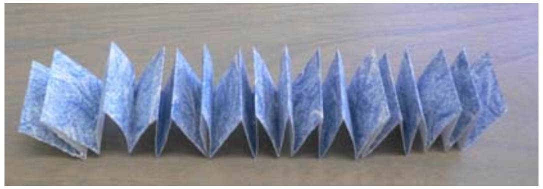
じゃばら折り

- じゃばら折り(1次元折り紙)

「中央で半分に折る」ことが
もっとも多くの折り目を
つけることができる

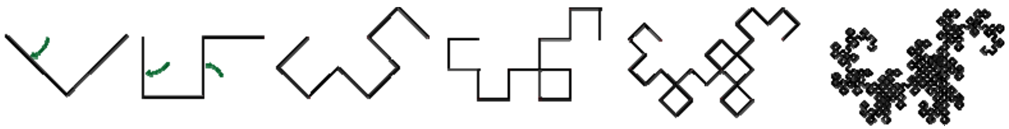
- 単純な方法： n 回折ればよい
- n 個の折り目をつけるには $\log n$ 回は折らないといけない

- もっと効率よく折り目をつけることはできるのか…?
- 一般のM/Vパターンに対してはどうか？



- CCCG 2008 の “Open Problem Session” にて
  上原が提案（Ron Graham が絶賛してくれた!!）
- 何度かの発表と結果の改善を経て多くの共著者を
  得て論文に至る

ドラゴン曲線



じゃばら折りの複雑さ

モデル：紙の厚みは0

[最大の疑問] n 個の折り目からなるじゃばら折りは
n 回折らなければ作れないのか？

最初の「大発見」であり，
研究を始めるきっかけと
もなるアルゴリズム！

1. 答えは“No”!
   - どんなM/Vパターンでも高々 $\lfloor n/2 \rfloor + \lceil \log n \rceil$ 回折れば作れる！
   - では $o(n)$ 回ではどうか?
   - Yes!! …わずか $O(\log^2 n)$ 回の折りで作る方法がある

2. 下界; $\log n$
   - じゃばら折りは $\Omega(\log^2 n/\log\log n)$ 回未満の折りでは作れない!!

一般化じゃばら折りの複雑さ

[次の疑問] 与えられた長さ n の任意の M/V パターンを折る問題の複雑さはどうか？

証明技法が面白い

□ **どんなM/Vパターンでも高々 $\lfloor n/2 \rfloor + \lceil \log n \rceil$ 回折れば作れる!!**

1. 上界:
どんなM/Vパターンも $(4+\varepsilon)\dfrac{n}{\log n} + o\left(\dfrac{n}{\log n}\right)$ 回の折りで作れる

計算量のオーダーは同じ

2. 下界:
ほぼすべてのM/Vパターンは $\dfrac{n}{3+\log n}$ 回以上折らないと作れない

[つまり] じゃばら折りは例外的に簡単なパターンであった！

•定義：単位長折り問題

2種類のパリティがあるところが{難しい|面白い}
- “表/裏”は層のパリティで決まる
- 同じパリティでないと重ねて折れない

入力: 長さ $n+1$ の紙と $\{M, V\}^n$ の文字列 s
出力: s にしたがって等間隔に折り目のついた紙

基本操作
1. 整数点で{一部/全部}の紙を{山/谷}折りにして平坦にする(=単純折りモデル)
2. {すべて/任意/直前}の折り目で開く(=単純折りの逆操作)

規則(仮定)
1. それぞれの折り目は最後に折られた方向を記録している
2. 紙は折り目を除いて剛体

目標: 折り操作の回数の最小化

注意: 紙を開く操作のコストは気にしない

- 単位長折りの上界(1)

□ どんなパターンも $\lfloor n/2 \rfloor + \lceil \log n \rceil$ 回で折れる

1. 指定にしたがって紙を中心で半分に折る
2. 折られた紙の中心を見て、MとVの多数決をとる
 (裏返った紙に注意)
3. 多数決に従って中心で半分に折る
4. 2と3を繰り返す
5. 全部広げる
6. 間違った折り目を逐一直す

ステップ1~4の折りは $\log n$ 回でステップ6は高々 $n/2$ 回の折り

□

じゃばら折りの上界(1)

[観測]
もし $f(n)$ 回の折りで「n 個の山折り」が作れるなら、
n 個のじゃばら折りは $2\ f(n/2)$ 回の折りで作れる

以下の方法でよい:

- 偶数点だけに着目して $f(n/2)$ 回折って全部山折りにする;
- 紙を裏返す;
- 奇数点だけに着目して $f(n/2)$ 回折って全部山折りにする.

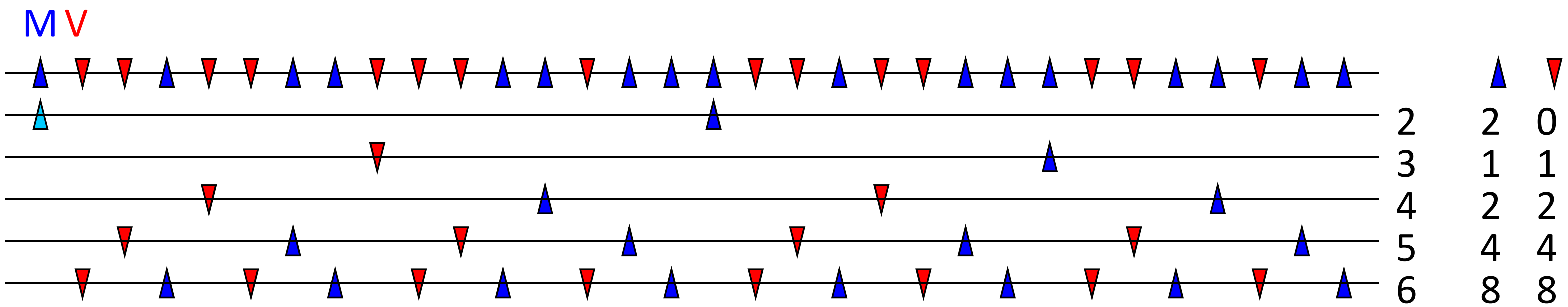
よって以下では「山折り問題」を考える

参考:「山折り問題」のO($n^{0.69}$)アルゴリズム

[定理] 山折り問題を解くO($n^{0.69}$) アルゴリズム

[証明] $n=2^k$ として以下のアルゴリズムを使う

1. 左端を山折りにして、長さを 2^k-1 にする
2. 中央で山折りにする
3. 長さ1になるまで2を繰り返す
4. 全部開いて…

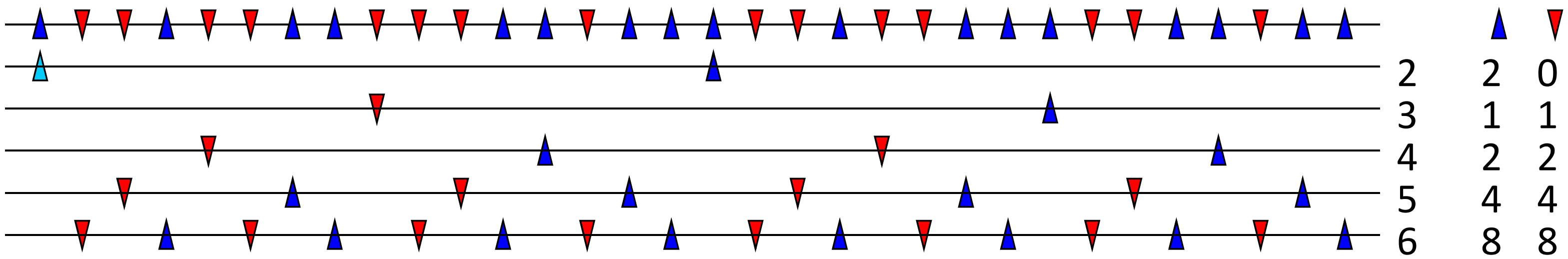


k+1回折って $2^{k\text{-}1}$+1 個の山と $2^{k\text{-}1}$-1 個の谷ができる

参考:「山折り問題」の$O(n^{0.69})$アルゴリズム

[定理] 山折り問題を解く$O(n^{0.69})$ アルゴリズム

[証明]



$2^{k\text{-}1}$-1 個の谷は独立で等間隔な k-1 個の層に分けられる!!

それぞれ独立に再帰的に解ける！

$$\therefore \quad f(2^k) = 1 + k + f(2^{k-2}) + f(2^{k-3}) + \cdots + f(4) + f(2) + f(1)$$

$$f(2^{k-1}) = k + f(2^{k-3}) + \cdots + f(4) + f(2) + f(1)$$

$$f(2^k) - f(2^{k-1}) = f(2^{k-2}) + 1$$

$$(f(2^k) + 1) = (f(2^{k-1}) + 1) + (f(2^{k-2}) + 1)$$

k に関するフィボナッチ数列！

参考:「山折り問題」のO($n^{0.69}$)アルゴリズム

[定理] 山折り問題を解くO($n^{0.69}$) アルゴリズム

[証明]

$$(f(2^k)+1) = (f(2^{k-1})+1) + (f(2^{k-2})+1)$$

k に関するフィボナッチ数列!

[フィボナッチ数列]

$F_0=0,\ F_1=1,\ F_i=F_{i-1}+F_{i-2}\ (i>1)$

0,1,1,2,3,5,8,13,21,34,…

$$F_i = \frac{1}{\sqrt{5}}\left(\left(\frac{1+\sqrt{5}}{2}\right)^i - \left(\frac{1-\sqrt{5}}{2}\right)^i\right)$$

初期条件:

$$f(2^0)=1,\ f(2^1)=2,\ f(2^2)=4$$

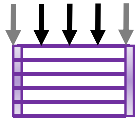
よって

$$f(2^k)+1 = F_{k+3}$$

$$\therefore\ f(n) = f(2^k) = \frac{1}{\sqrt{5}}\left(\left(\frac{1+\sqrt{5}}{2}\right)^{k+3} - \left(\frac{1-\sqrt{5}}{2}\right)^{k+3}\right) - 1 \sim \frac{1}{\sqrt{5}}\left(\frac{1+\sqrt{5}}{2}\right)^{k+3}$$

$$= O\left(\left(\frac{1+\sqrt{5}}{2}\right)^{\log n}\right) = O\left(n^{\log\frac{1+\sqrt{5}}{2}}\right) = O(n^{0.694242}) \quad \square$$

山折り問題を $\log^2 n$ 回の折りで解くアルゴリズム



ステップ1;

1. 「半分に折る」を繰り返して以下をえる ($\log n$-2 回の折り): [vvv]
2. 3回山折りして以下を得る: [MMM]
3. 開く; vMMMvvvvvMMMvvvvvMMMvvvvvMMMvvvvv...

ステップ2;

[MvvvvvM]

1. すべての “vvvvv”が重なるように半分折りを繰り返す($\log n$-3 回折る)
2. 5回山折りして以下を得る: [MMMMMMM]
3. 開く; vMvMMMMMMMvMvvvvvMvMMMMMMMvMvvvvvMvM

ステップ3; “vvvvv” が一つ残るまでステップ 2 を繰り返して以下を得る:

vMvMMMvMMMMvMMMMMMMvMMMvMMMMvMvvvvvMvM

ステップ4; すべてのとびとびの v を一つずつ折る.

- ステップ2~3の繰り返し回数; $\log n$
- ステップ4での v の個数; $\log n$

折りの回数のトータル~ $(\log n)^2$

単位長折り問題の下界

[定理] o(2^n)個の例外を除くほぼすべてのパターンは
$\Omega(n/\log n)$ 回折らないと作れない

{表/裏}×{前/後}

[証明] 単純な数え上げ法による:

- n 個の折り目のパターンの数 $> 2^n/4 = 2^{n-2}$
- k 回折ったことで作れるパターンの数 $<$

山/谷

$(2\times n)\times(n+1)\times(2\times n)\times(n+1)\times\ldots\times(n+1)\times(2\times n)$

位置

可能な展開

$<(2n(n+1))^k$

- よって以下が成立する場合、高々 k 回の折りではすべてのパターンは作れない: $\sum_{i=0}^{k}(2n(n+1))^i \le (2n(n+1)+1)^k < 2^{n-2}$

ここで $n \ge 2, k = O\left(\frac{n}{\log n}\right)$ とおくと $(2n(n+1)+1)^k = o(2^n)$ をえる。

□

任意のパターンを $cn/\log n$ 回の折りで作るアルゴリズム（概要）

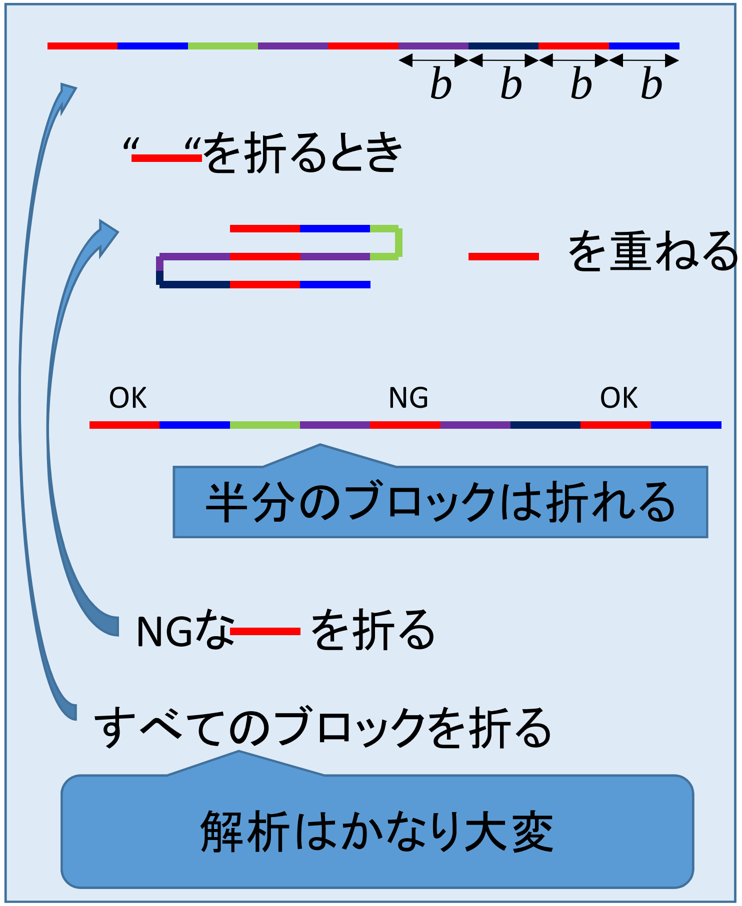
nに応じて適切に b を選ぶ.

準備：

- パターンを大きさbのブロックに分割;
  1. 各ブロックは $O(n/\log n)$ 回で折れる
  2. 2^b はそれほど大きくない

アルゴリズム：

- 可能なそれぞれのブロックパターンに対して
  1. 同じパターンをもつブロックが重なるように半分折りする
  2. 裏返っているブロックを直す
  3. 重ねるために折った所を直す



未解決問題

- じゃばら折り
  - 上界 $O(\log^2 n)$ と下界 $\Omega(\log^2 n / \log\log n)$ を近づける
- 「ほぼすべてのパターンは難しい」と言うけれど…
  - $(cn/\log n)$ 回の折りが本当に必要な**具体的な M/V パターン**の構成方法はわかっていない
- 紙を開くコストは無視しているけど…
  - 「折る回数」＋「開く回数」を最小化するとよいかも
  - (たかだか折った回数しか開けないけど…)
- 一般の間隔や2次元への拡張も…

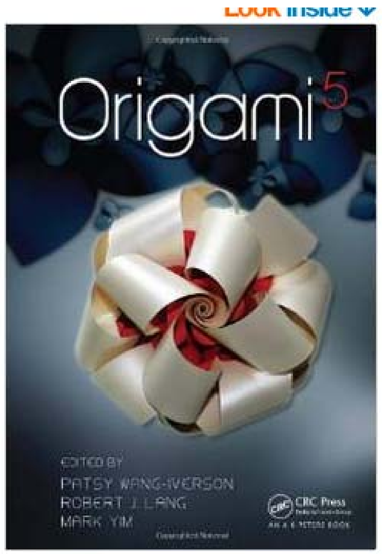
「折り」のアルゴリズムと計算量の関係

5. 時間計算量
   - “Folding complexity”入門
     - 理論上、世界最速のジャバラ折りアルゴリズム

6. <u>領域計算量（？）</u>
   - <u>切手折り問題</u>
   - 折り目幅最小化問題
     - NP完全問題、FPTアルゴリズムなど
     - これも予想以上にコンピュータサイエンス！

7. （折り紙における決定不能問題）
   - 対角線論法と不完全性定理

切手折り問題



上原隆平（うえはらりゅうへい）
北陸先端科学技術大学院大学　情報科学研究科
Japan Advanced Institute of Science and Technology (JAIST)
uehara@jaist.ac.jp
http://www.jaist.ac.jp/~uehara

主要な論文

Ryuhei Uehara, Stamp foldings with a given mountain-valley assignment in *ORIGAMI*^{5}, pp. 585-597, CRC Press, 2011.

一般化じゃばら折り問題

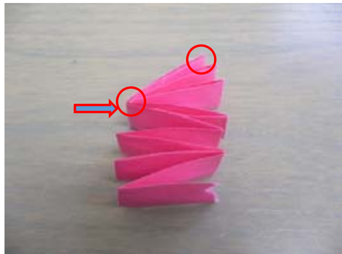
- 入力：等間隔の山折りと谷折りの列
- 出力：入力を実現する平坦な折りたたみ状態

[定理] どんな入力に対しても、平坦な折りたたみ状態が存在する。

[証明(?)] 端から一つずつ折っていけばよい。

[疑問] 「良い平坦折り状態」とは？？

例：山谷山谷山谷山山谷山谷山谷山



左側の一つの折り目に右側を全部入れる

× 良くない！！

どの折り目も2枚以下の紙しか挟まっていない

○ 良い！！

- 平坦折りの「良さ」

[疑問]「良い平坦折り状態」とは？？

ある折り目の幅
＝その折り目に挟まった
他の紙の枚数

「折り目に挟まった紙」が少ないほど「良い」！

- 厚みがあっても精度が確保されやすい
- バランスがよさそう
- 「時間」と「伸び」はトレードオフがある
（計算機モデルにおける「時間」「領域」に似ている）

良さの指標

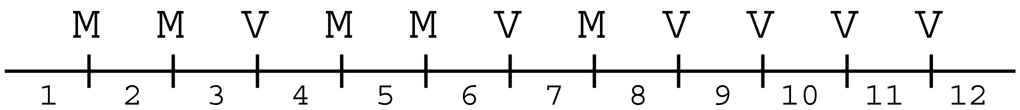
1. 折り目の幅の最大値
2. 折り目の幅の平均値
3. 折り目の幅の総和

指標[2]と[3]は本質的に同じ:
平均値＝(総和/折り目の数)

指標による結果の違い

答えが自明ではない例

入力: 山山谷山山谷山谷谷谷谷

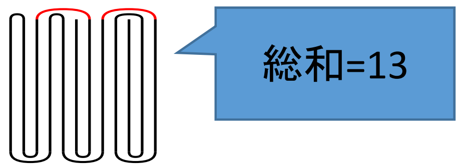


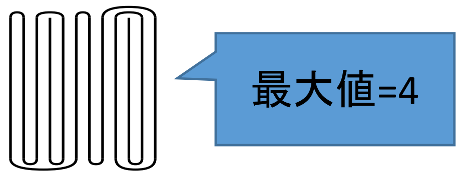
折りたたみ方: 正当な平坦折りの個数:100通り

解答: 指標によって結果が違う(しかも両方唯一解):

- 最大値が最小値3の唯一解
  [5|4|3|6|7|1|2|8|10|12|11|9]

- 総和が最小値11の唯一解
  [5|4|3|1|2|6|7|8|10|12|11|9]





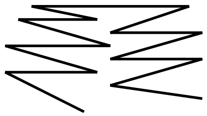
折り目幅最小化問題

入力: 長さ $n+1$ の紙と[山/谷]の長さ n の文字列 s
出力: s に従って平坦に折られた紙
目的: 折り目幅の少ない「良い」平坦折り状態

- わかっていること;
  1. ２種類の最適化問題の解答は違う場合がある
  2. パターンがじゃばら折りである⇔
伸びが0である⇔
折り方が一意的

     [証明]　余白がせますぎて書けない

  3. 指数関数的な組合せをもつ入力例がある



[例]　山谷山谷山谷...山谷山山谷山谷山谷...山谷山

折り目幅最小化問題

当時解けなかった問題:

- NP完全(後年解決！)
- 「幅≦ k 」としたら多項式時間で解けるか?
  (これも後年解決！)
- もっとも組合せの多いパターンは?（未解決）

あとで紹介

このとき解けた問題:

1. ランダムなパターンの折りたたみ方の期待値
2. （「単純折りモデル」の万能性）

折り目幅最小化問題

このころ解けた問題：

1. ランダムなパターンの折りたたみ方の期待値
   - 長さ n のランダムな山谷パターンに対する折りたたみ方 $f(n)$ が相対に小さければ、力技で解けるかもしれない…

   …という考えは甘かった

   - 実験的: $f(n)=\Theta(1.65^n)$
   - 理論的: $f(n)=\Omega(1.53^n)$ かつ $f(n)=O(2^n)$

   …単純な全数チェックは望みがない

2. （「単純折りモデル」の万能性）

折り目幅最小化問題

1. ランダムなパターンの折りたたみ方の期待値

- 長さ n の紙を（折り目は考えずに）長さ1になるまで折りたたむ方法を $F(n)$ とすると、$f(n)=F(n)/2^n$

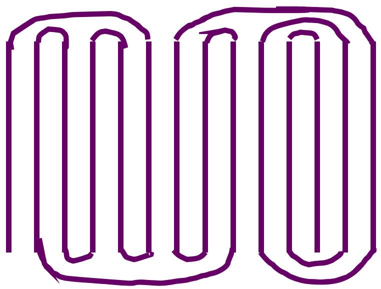
- $F(n)=\Omega(3.06^n)$ かつ $F(n)=O(4^n)$ を示せばよい。

後日談：この関数$F(n)$は有名な未解決問題
「切手折り問題(stamp folding)」として知られており，
既存の上下界は $\underline{2^n \leqq F(n)} \leqq 4^n$ であった！

切手折り問題

$F(n)$の上界: $F(n)=\mathrm{O}(4^n)$

[証明] 満たすべき条件：奇数折り目と偶数折り目がそれぞれ入れ子状になっていなければならない



入れ子

(()())(()(()))

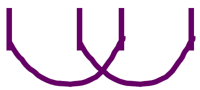
$n/2$組の()の組合せ
＝カタラン数$C_{n/2}$

×

入れ子

·(()()())((·))

$n/2$組の()の組合せ
＝カタラン数$C_{n/2}$



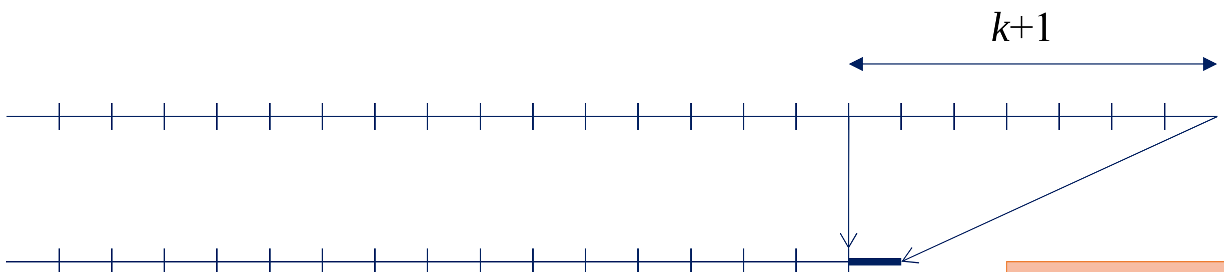
ダメ！

連結性は考慮していない

切手折り問題

$F(n)$の下界: $F(n)=\Omega(3.07^n)$

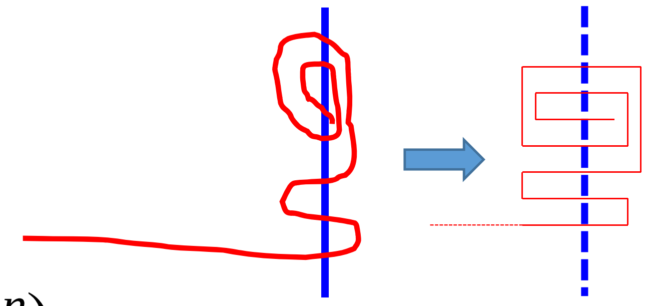
[証明]「紙の最後の $k+1$ 長を折ってのり付け」を考える;



- $F(n)$: 長さ $n+1$ の紙の折りたたみ方
- $g(k)$: 長さ $k+1$ の紙の折りたたみ方
  ただし左端点は覆われない

とすると次を得る: $$f(n) \ge (g(k))^{\frac{n}{k-1}} = \left(g(k)^{1/(k-1)}\right)^n$$

n は<u>変数</u>だけど
k は<u>定数</u>



切手折り問題

$F(n)$の下界: $F(n)=\Omega(3.07^n)$

[証明]「紙の最後の $k+1$ 長を折ってのり付け」を考える;

- $g(k)$: 長さ $k+1$ の紙の折りたたみ方（ただし左端点は覆われない）

とすると

$g(k)$=“the number of ways a semi-infinite directed curve can cross a straight line k times”

by The On-Line Encyclopedia of Integer Sequences (OEIS;A000682)

上記のデータベースによると、$g(43)=830776205506531894760$.

よって

$$f(n) \geq (g(k))^{\frac{n}{k-1}} = \left(g(k)^{1/(k-1)}\right)^n$$

に代入すればOK!

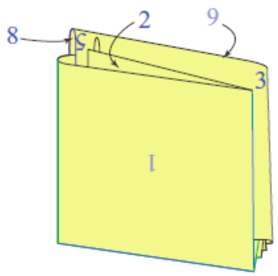
切手折り問題の拡張

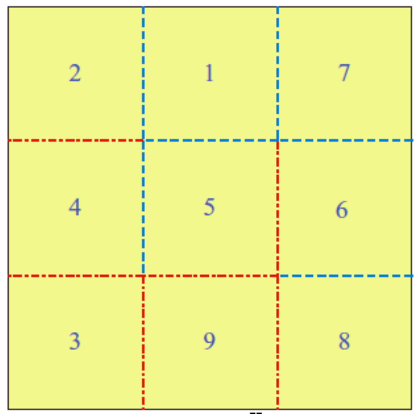
２次元への拡張～地図折り問題

入力：2次元の山折り/谷折りのパターン

出力：1×1に「折り畳んだ状態」が存在するか？

- m×nの地図折り問題の困難性は未解決
- 2×nの地図折り問題の多項式時間アルゴリズムは [Morgan 2012]で与えられたが… O(n^9)時間
- すごく難しいパズル：





「折り」のアルゴリズムと計算量の関係

5. 時間計算量
   - “Folding complexity”入門
     - 理論上、世界最速のジャバラ折りアルゴリズム

6. 領域計算量（？）
   - 切手折り問題
   - 折り目幅最小化問題
     - NP完全問題、FPTアルゴリズムなど
     - これも予想以上にコンピュータサイエンス！

7. （折り紙における決定不能問題）
   - 対角線論法と不完全性定理

A survey on computational complexity of finding good folded state with few crease width

Ryuhei Uehara (JAIST, Japan)

- Erik D. Demaine, David Eppstein, Adam Hesterberg, Hiro Ito, Anna Lubiw, Ryuhei Uehara and Yushi Uno: Folding a Paper Strip to Minimize Thickness, *The 9th Workshop on Algorithms and Computation (WALCOM 2015)*, Lecture Notes in Computer Science Vol. 8973, pp. 113-124, 2015/02/26-2015/02/28, Dhaka, Bangladesh.
- Takuya Umesato, Toshiki Saitoh, Ryuhei Uehara, Hiro Ito, and Yoshio Okamoto: Complexity of the stamp folding problem, Theoretical Computer Science, Vol. 497, pp. 13-19, 2012.

関連研究？

- 英語のイデオムで「紙を10回半分に折る」で「できないことの例え」になるらしい。

Folding Paper in Half 12 Times:
The story of an impossible challenge solved at the Historical Society office

Alice laughed: "There's no use trying," she said; "one can't believe impossible things." "I daresay you haven't had much practice," said the Queen.

Through the Looking Glass by L. Carroll

BRITNEY'S FOLDING RECORD STILL HOLDS

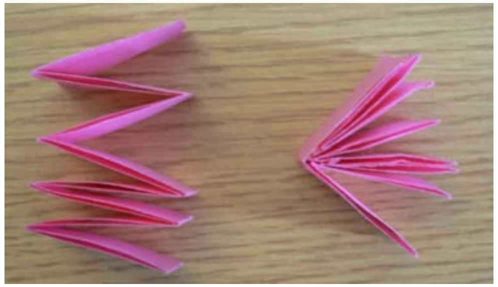
The long standing challenge was that a *single* piece of paper, no matter the size, cannot be *folded* in half more than 7 or 8 times. Recently, reports have been made that Britney's paper folding record of folding a piece of paper in half 12 times has been broken. These current attempts, though laudable and will eventually be

Photo of the 11th Fold, One More to go.

Minimization of Crease width

Good!

Bad!



Input: Paper of length $n+1$ and $s \in \{M, V\}^n$

Output: folded paper according to s

Goal: Find a *good* folded state with *few* *crease width*

- At each crease, the number of paper layers between the paper segments hinged at the crease is *crease width* at the crease
- Two minimization problems;
    - minimize maximum
    - minimize total (=average)

No!!

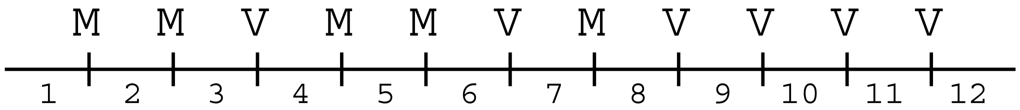
It seems simple, … so easy??

Cf. I’d checked that by brute force…

Crease width problem

Simple non-trivial example (1)

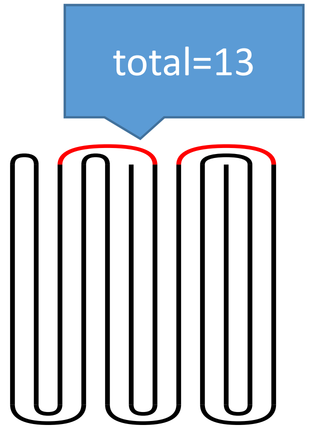
Input: MMVMMVMVVVV



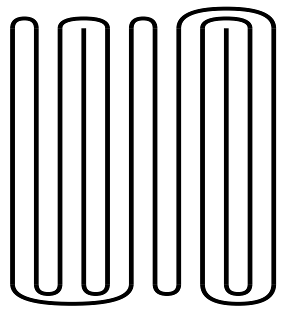
The number of feasible folded states : **100**

Goal: Find a *good* folded state with few *crease width*

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



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

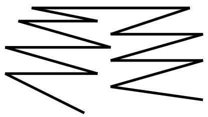


Crease width problem

Simple non-trivial example (2)

A few facts;

- a pattern has a unique folded state iff it is pleats
- solutions of {min max} and {min total} are different depending on a crease pattern.
- there is a pattern having *exponential* combinations



This pattern is almost pleats, but it has exponentially many folded states....

Main Results

The crease width problem (unit interval)

- Min-Max problem: NP-complete

  (Reduction from 3-Partition)

It is intractable even for small n...

- Min-Total problem:
  - Complexity is *still open*...
  - *Fixed Parameter Tractable algorithm*;

    it runs in $O((k+1)^k n)$ time, where k is the total crease width.

    the algorithm itself is natural, but analysis is bit tricky.

It is solvable if k *is* quite small...

MinMax is NP-complete

Proof: Polynomial time reduction from 3-Partition.

<u>3-Partition</u>:

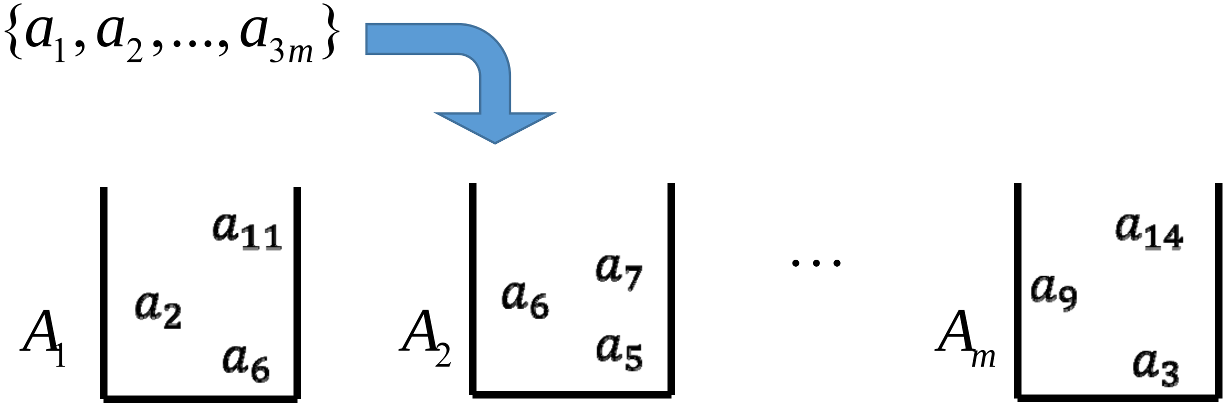
$(B/4 < a_j < B/2)$

Input: Set of integers $A = \{a_1, a_2, \ldots, a_{3m}\}$ and integer B

Question: Is there a partition of A to $A_1, \ldots, A_m$

such that $|A_i| = 3$ and $\sum_{a_j \in A_i} a_j = B$

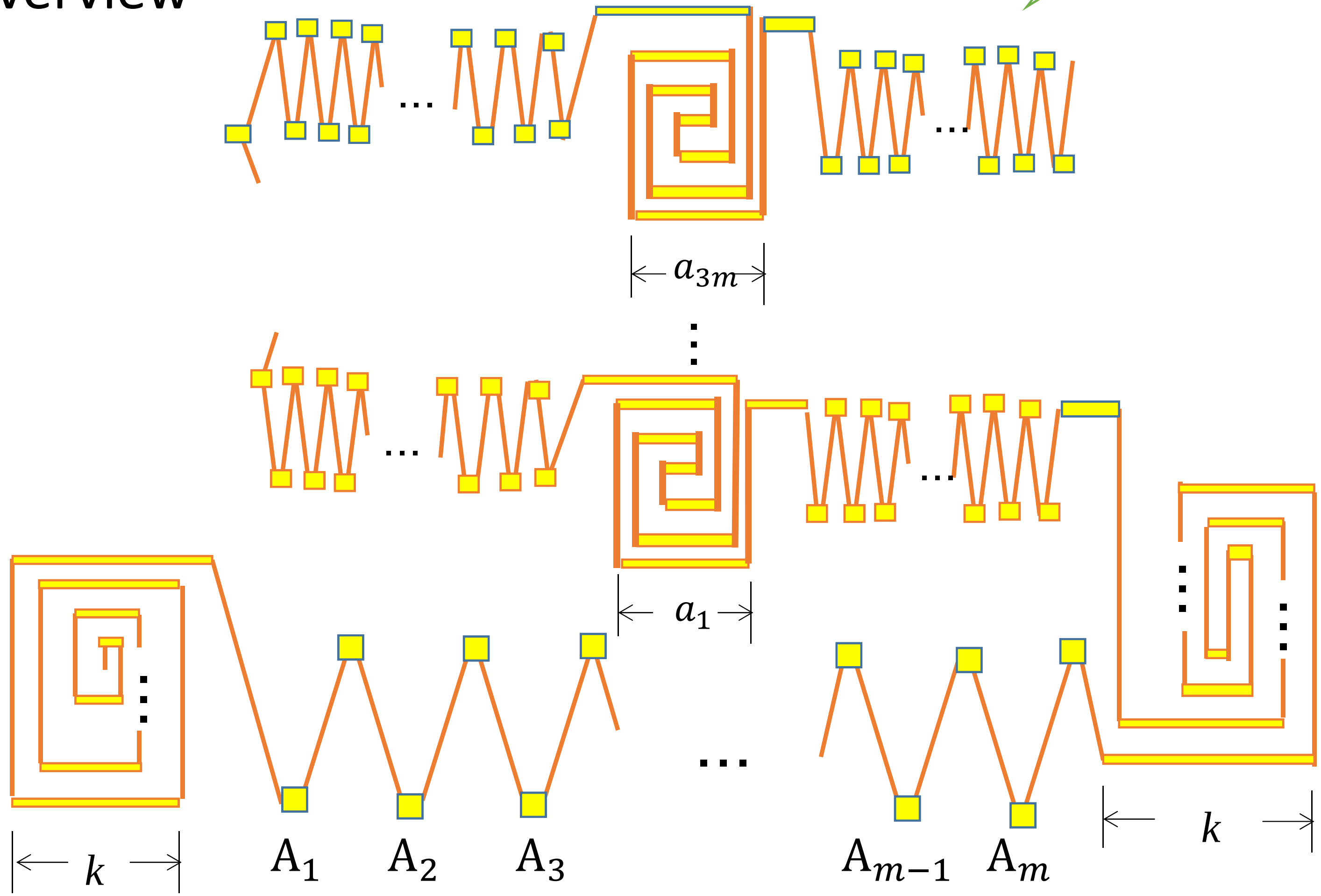
$A = \{a_1, a_2, \ldots, a_{3m}\}$



Construction for MinMax

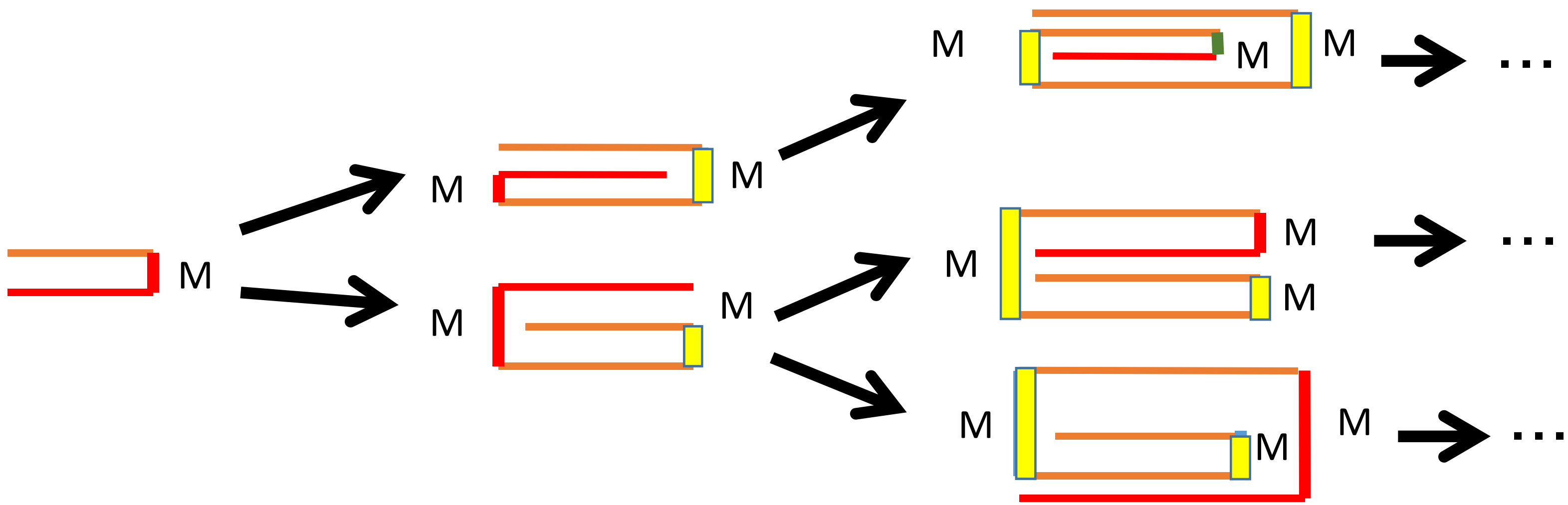
Overview

図がわかりにくいのであとで再放送。。。



(Poly-time) Algorithm for MinTotal

- Enumerate all folding ways with respect to the string s up to total crease width k.
- Each folded state is generated incrementally.



- Check the total crease width at each increment.

Running time

- The algorithm for given fixed total crease width k runs in $O(n^{2+k})$ time.

  at each crease, the sequence of c.w. is

  - $c_1, c_2, \ldots, c_i$, ... with $\sum_{i=1,2,\ldots} c_i \leq k$

  that is a partition of $\leqq k$
- With more careful (and complex) analysis shows that the algorithm runs in $O((k+1)^k\, n)$ time!!

That is, this is fixed parameter tractable algorithm!

Known results

最近ここで進展がありました。

Possible extensions in 2012;

1. Non-unit intervals between creases

   How can you measure the thickness of pile of various lengths?

2. 2-Dimenaional (…related to Map-folding?)

   How can you measure the crease-width in 2D?

3. Different Criteria for “space complexity”

   We have few ideas…

   (by Prof. Iwama: you can fold left $<k$ and right$>k$ creases in $[1..n]$)

   (Area to fold for long-pipe folding)

Now we turn to・・・

- Quite simple model
  1. 1 dimensional
  2. Unit ~~length between~~ creases

Non-unit intervals!!

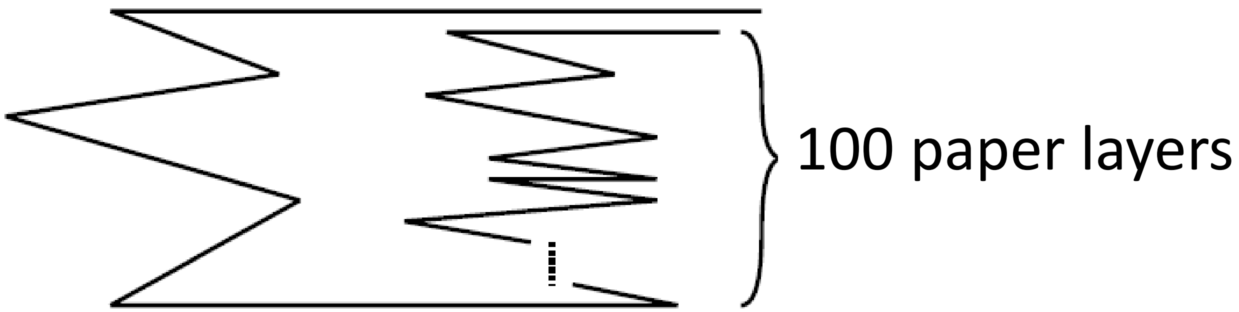
Not only M/V, but also lengths between creases are given

But,,, wait,,,

Erik D. Demaine, David Eppstein, Adam Hesterberg, Hiro Ito, Anna Lubiw, Ryuhei Uehara and Yushi Uno: Folding a Paper Strip to Minimize Thickness, *The 9th Workshop on Algorithms and Computation (WALCOM 2015)*

For non-unit interval creases…

Crease width = the number of paper layers at a crease?



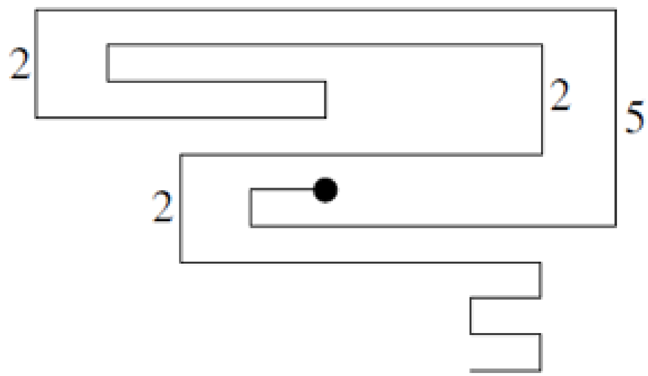
How can we count the paper layers?

For non-unit interval creases...

Crease width = the number of paper layers at a crease?

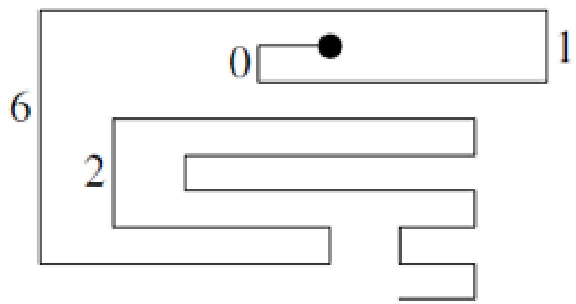
- We introduce three new "widths" of a folded state:
  - Two are natural extensions of Max-CW and Total-CW; one is totally new!
- For VMVMVVMMMM, e.g., we have;

Minimum
max crease width



$h = 11, m = 5, t = 11$

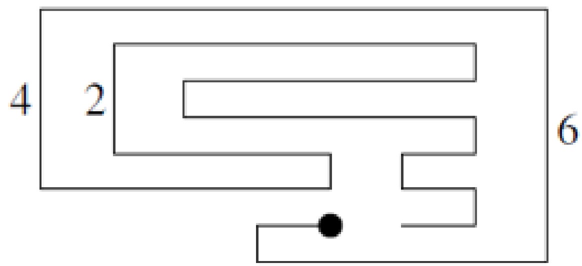
Minimum
total crease width



$h = 9, m = 6, t = 9$

New

Minimum **height**



$h = 8, m = 6, t = 12$

Main results in (DEHILUU 2015)

Summary

	Unit interval model in [USUIO2012]	General model in (DEHILUU 2015)
max crease width	NP-complete	⇒ NP-complete
total crease width	open	NP-complete [DEHILUU 2015]
height	trivial	NP-complete [DEHILUU 2015]

Proof Idea

Minimize height is NP-complete

Proof: Polynomial time reduction from 3-Partition.

<u>3-Partition</u>:

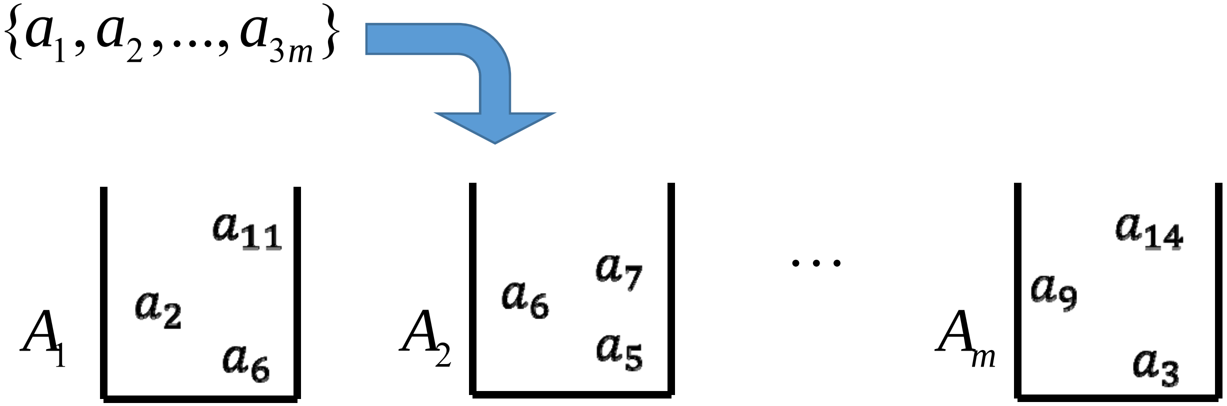
$(B/4 < a_j < B/2)$

Input: Set of integers $A = \{a_1, a_2, \ldots, a_{3m}\}$ and integer B

Question: Is there a partition of A to $A_1, \ldots, A_m$

such that $|A_i| = 3$ and $\sum_{a_j \in A_i} a_j = B$

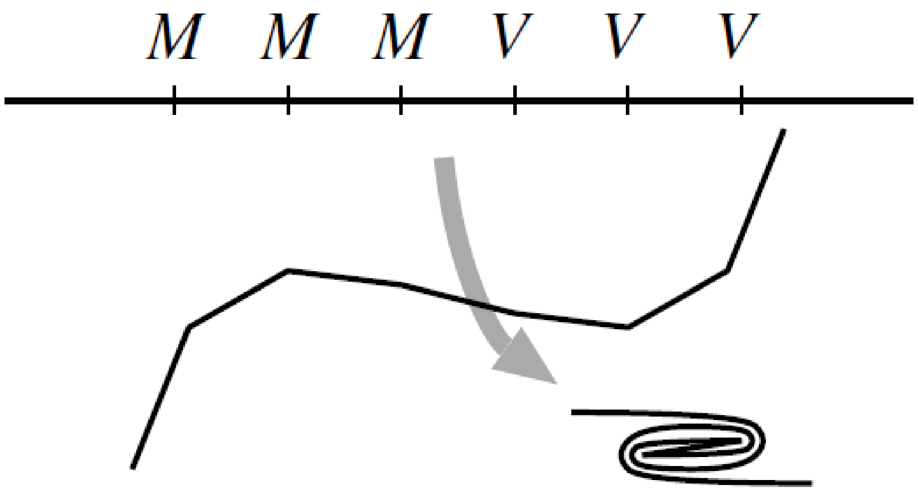
$A = \{a_1, a_2, \ldots, a_{3m}\}$



Minimize height is NP-complete

Proof: Polynomial time reduction from 3-Partition.

Basic gadget

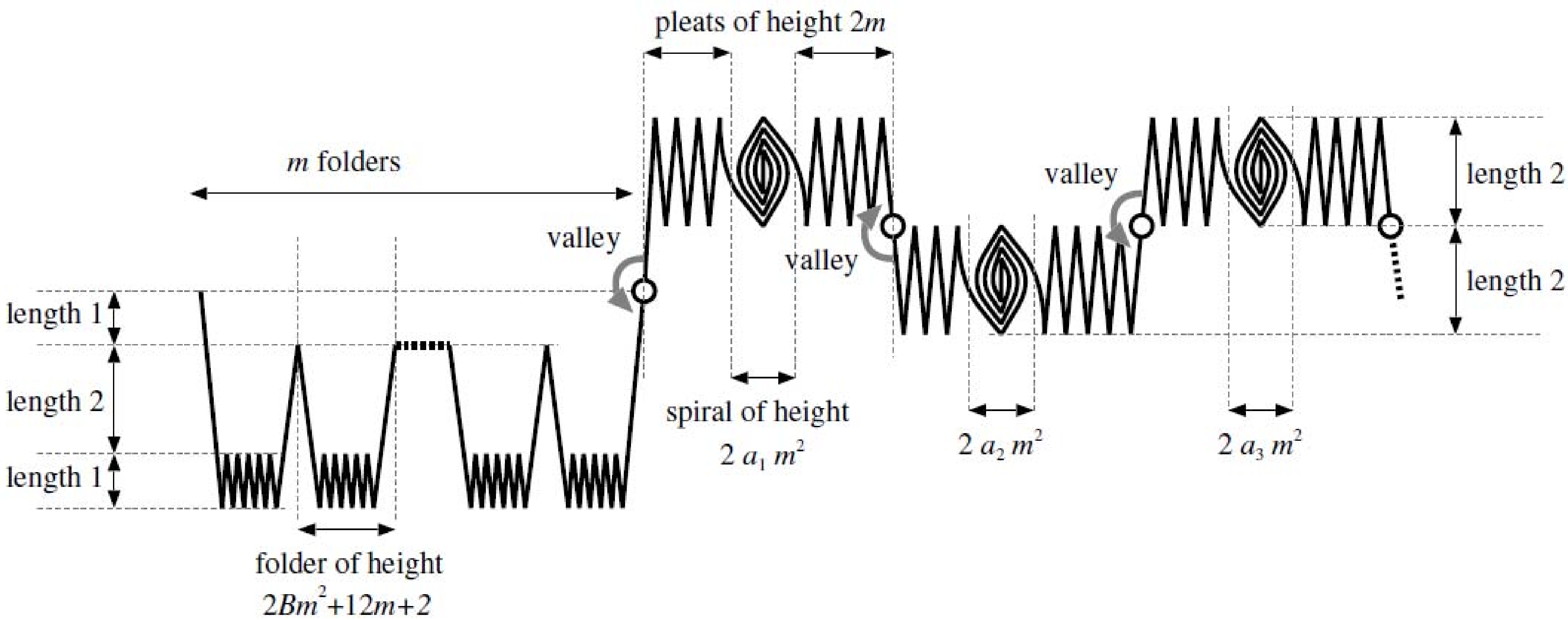


The way of folding is unique by bit longer end-edges

Minimize height is NP-complete

Proof: Polynomial time reduction from 3-Partition.

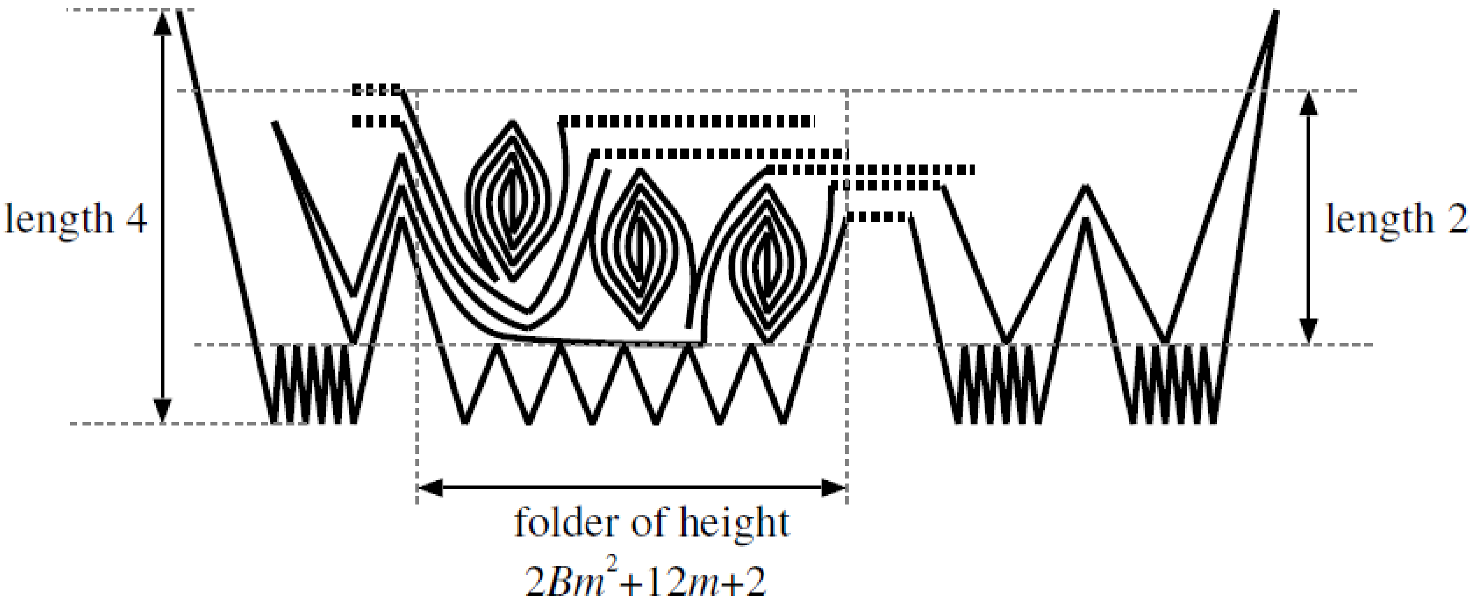
<u>Overview</u>



Minimize height is NP-complete

Proof: Polynomial time reduction from 3-Partition.

<u>Overview</u>



Summary & Future work…

Origami is interesting even in 1 dimension!!

	Unit interval model in [USUIO2012]	General model in (DEHILUU 2015)
max crease width	NP-complete	⇒ NP-complete
total crease width	open	NP-complete [DEHILUU 2015]
height	trivial	NP-complete [DEHILUU 2015]

Future work:

- Replace "open" into ???
- Extension to 2 dimension
    - Different measures of "thickness"?
- Estimation of the way of folding (~time complexity)
- Nicer model for "*Time-space trade off*" for Origami