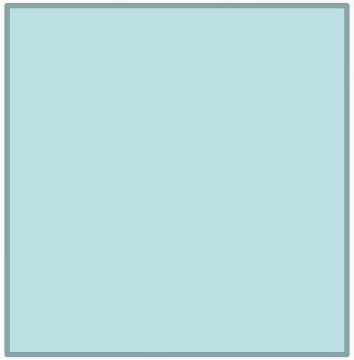
Complexity of the stamp folding problem

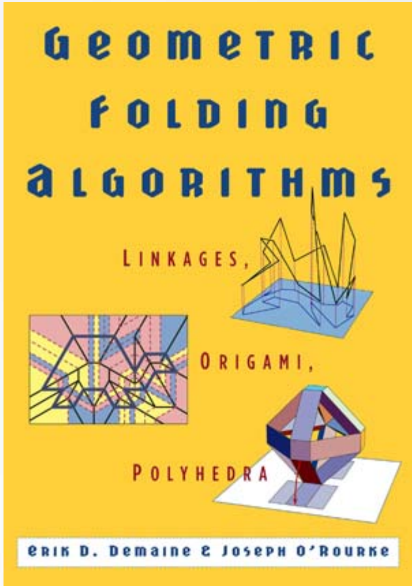
Takuya Umesato (JAIST)
Toshiki Saitoh (ERATO, JST)
Ryuhei Uehara (JAIST)
Hiro Ito (京都大学)
+Yoshio Okamoto (電気通信大学)

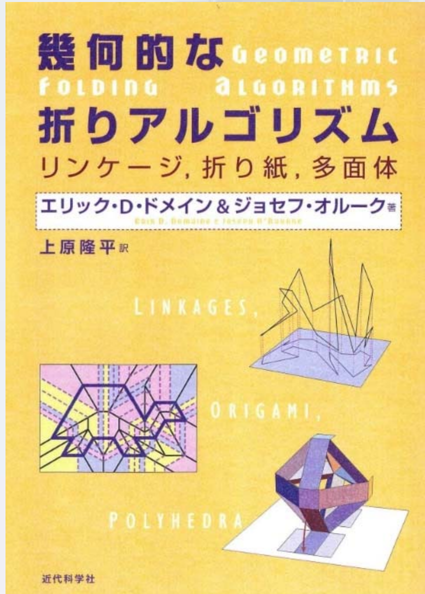
Introduction

- Computational Origami

- A sheet of paper







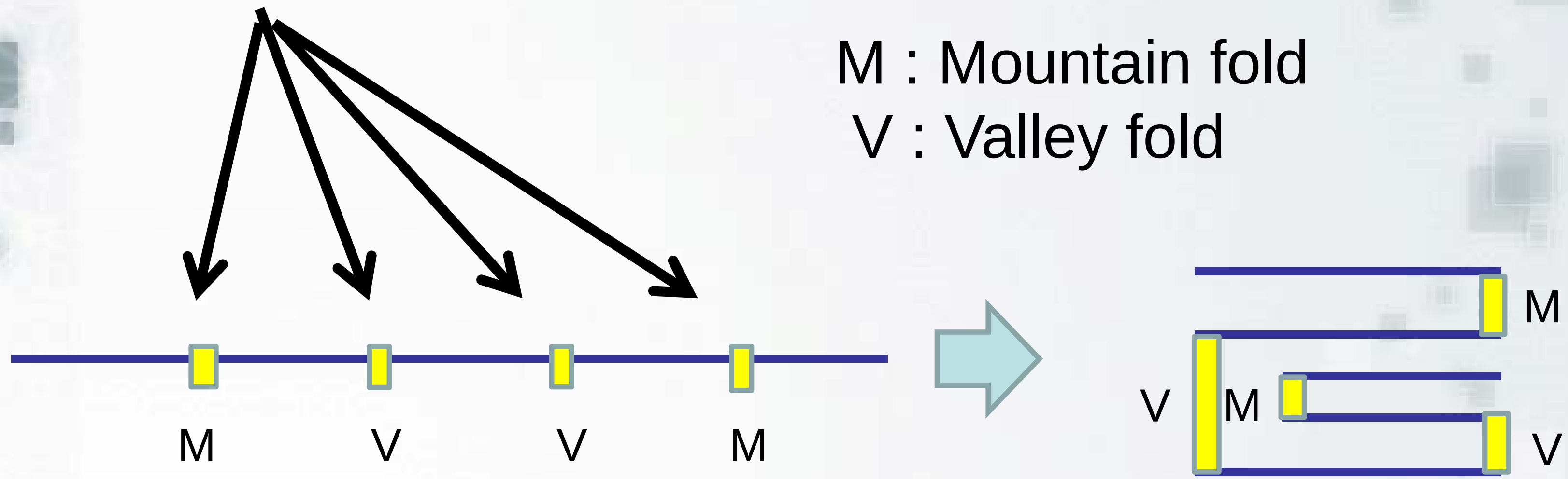
- In 2D, it is NP-hard to determine if a sheet of paper can be folded flat for a given crease pattern. [Bern and Hayes, 1996]

Introduction

- 1D paper (or long strip paper)

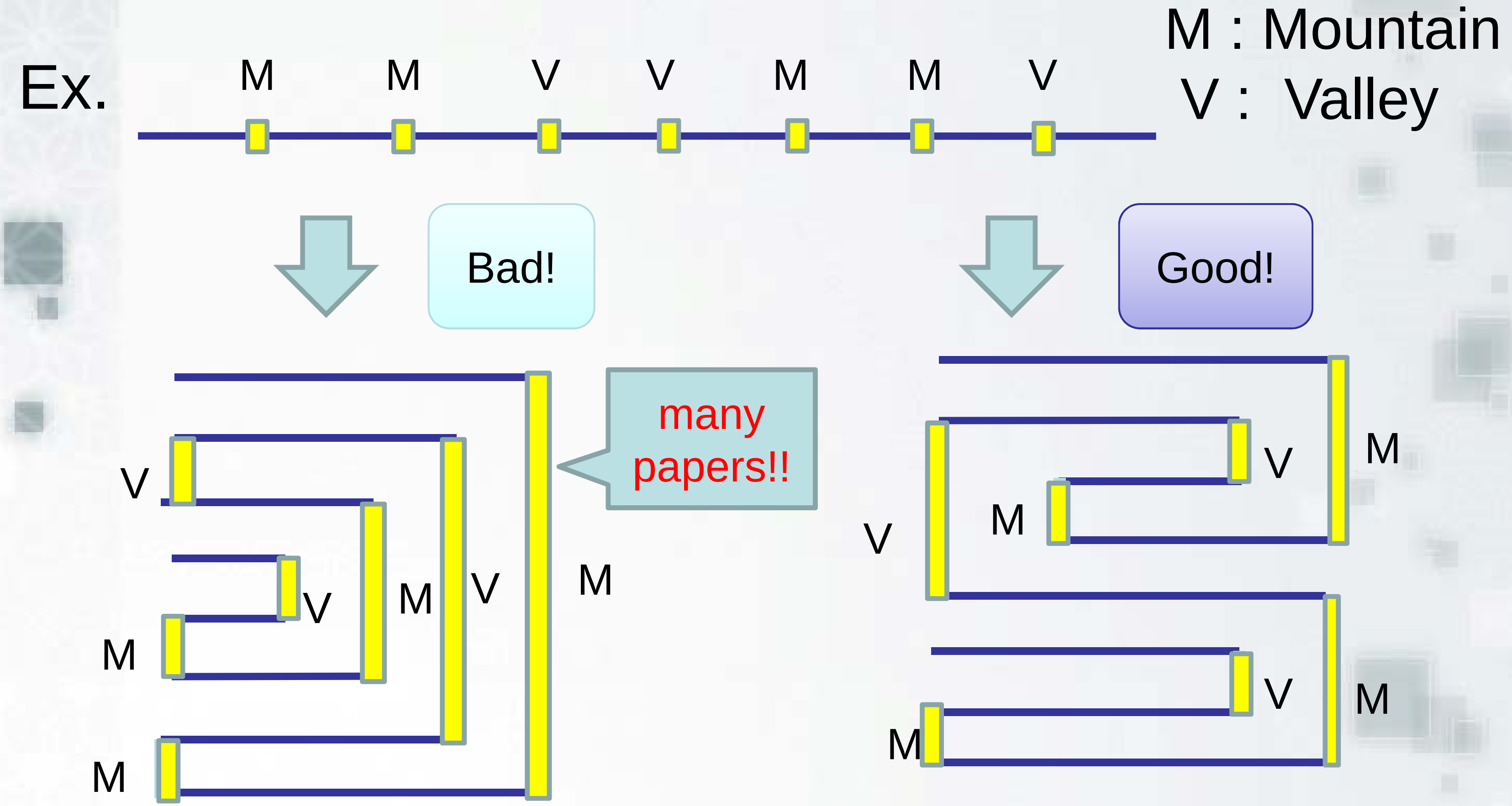
- Creases at unit intervals

M : Mountain fold
V : Valley fold



Introduction

- Given M/V pattern, fold it into unit length



New minimization problems

- Folding with least **crease width**

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

Output: folded paper according to s

Goal: Find a *best* folded state with small *crease width*

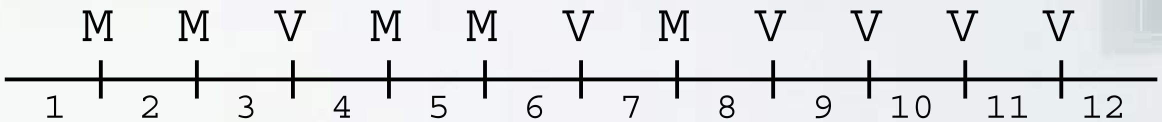
- At each crease, the number of papers between the papers hinged at the crease is *crease width.*
- Two minimization problems;
  - minimize maximum
  - minimize total (=average)

No!!

It seems simple, … so easy??

Simple non-trivial example

Input: MMVMMVMVVVV

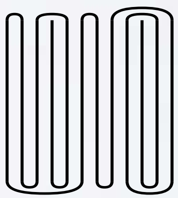


The number of feasible folded states : **100**

Goal: Find a *best* folded state with small *c.w.*

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

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



Stamp folding problem

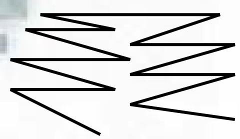
- Folding with least **crease width**

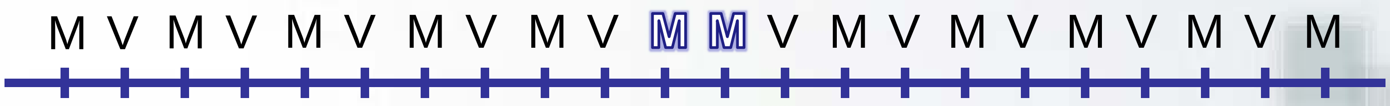
Input: Paper of length $n+1$ and $s \in \{M, V\}^n$

Output: folded paper according to s

Goal: Find a *best* folded state with small *crease width*

- Two criteria; MinMax and MinTotal
- A few facts;
  - solutions of MinMax and MinTotal are different depending on a crease pattern.
  - there is a pattern having exponential combinations





Stamp folding problem

Known result:

- If the crease pattern is given uniformly at random, the expected number of folding ways is exponential [Uehara, 2010].
  - so simple search does not work efficiently.

- Computational complexity of the stamp folding problem was open.

Main results

- MinMax : NP-complete

- MinTotal :
  restricted case can be solved
  in polynomial time.
  (If MinTotal $\leqq k$ for small constant k,
  it can be solved in poly-time.)

MinMax is NP-complete

Proof: Polynomial time reduction from 3-Partition.

<u>3-Partition</u>:

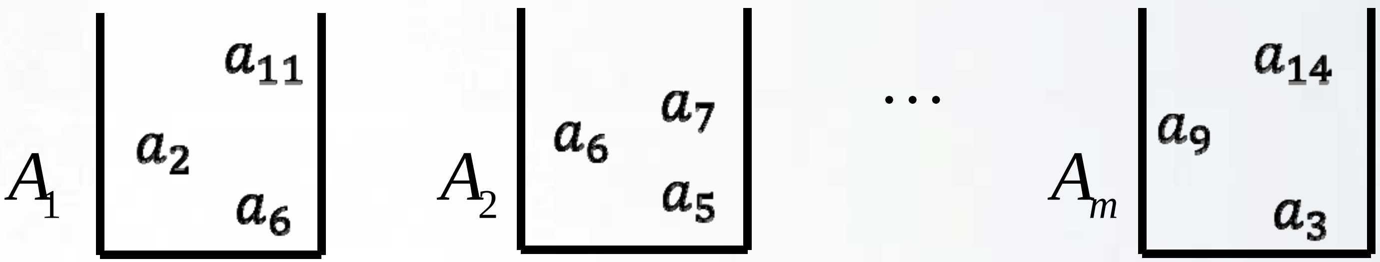
$(B/4 < a_j < B/2)$

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

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

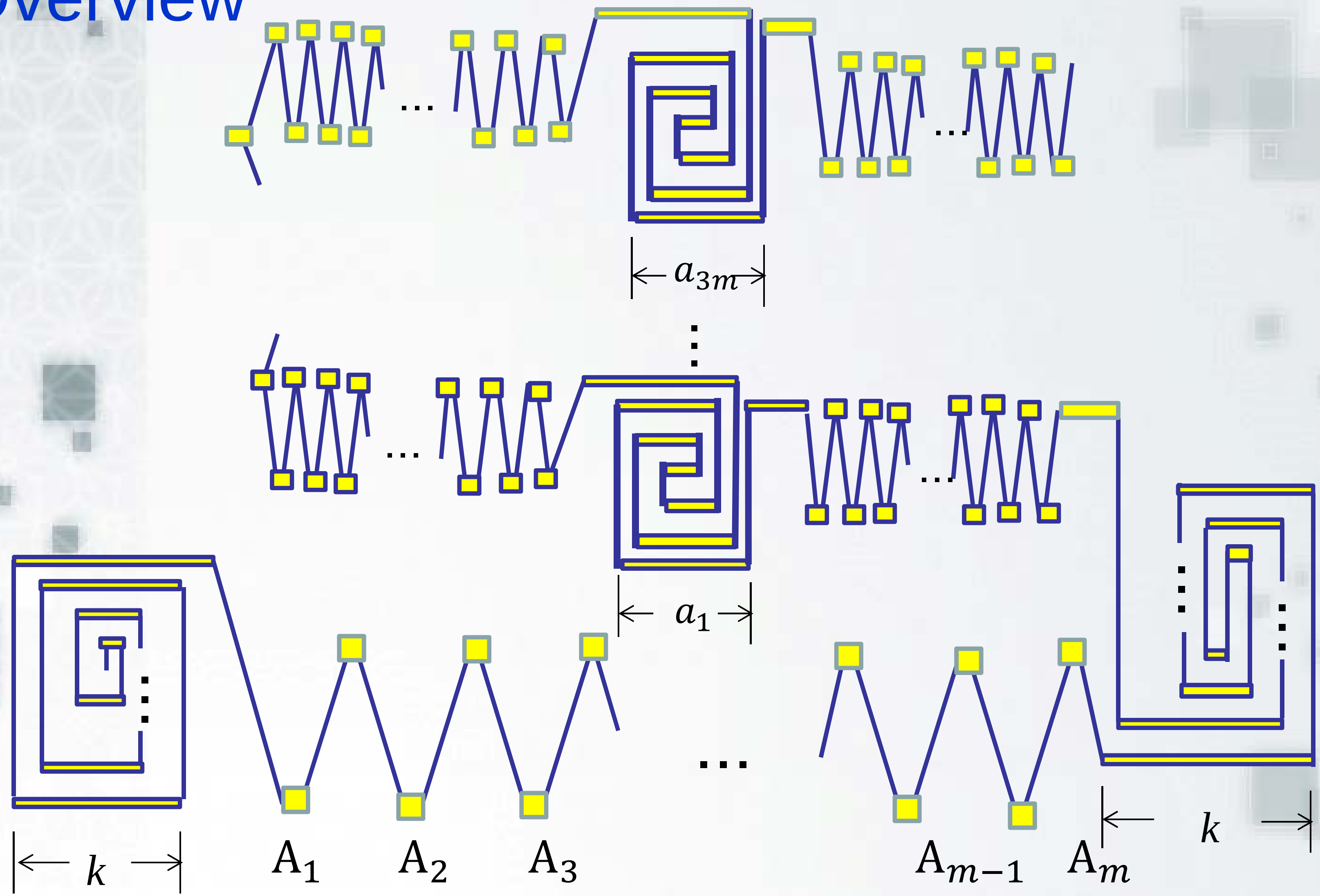
such that $|A_i|=3$ and $\sum_{a_j \in A_i} a_j = B$

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



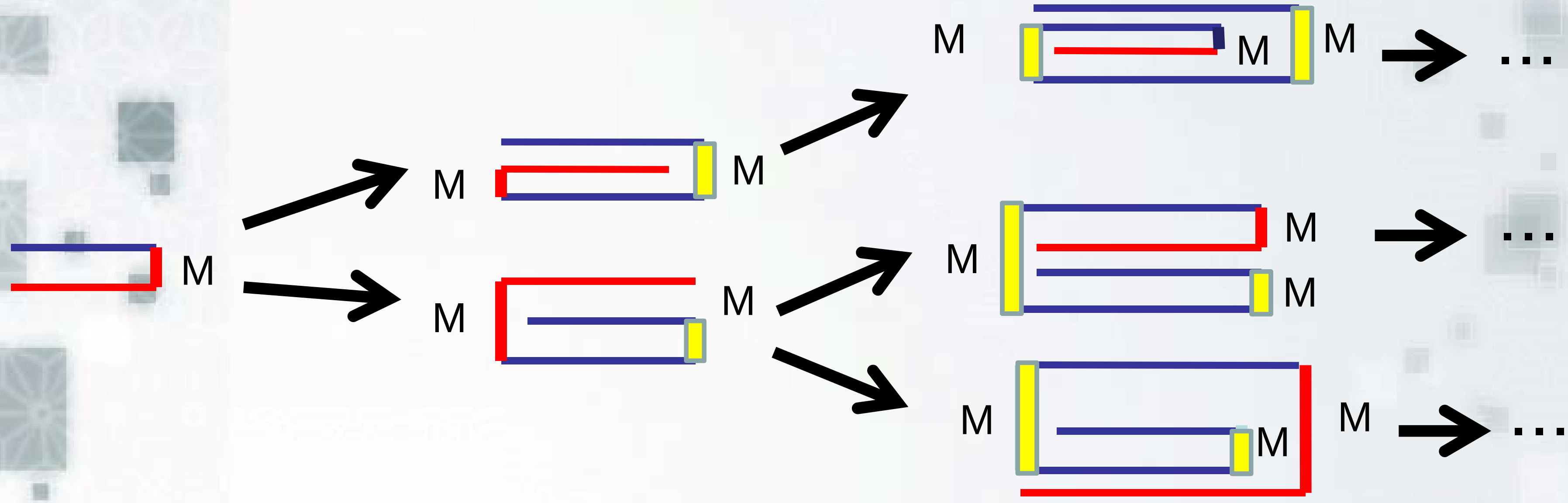
Construction for MinMax

Overview



(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
    - $c_1, c_2, \ldots, c_i, \ldots$ with $\sum_{i=1,2,\ldots} c_i \leq k$
  - that is a partition of $\leqq k$

Summary

- MinMax : NP-complete
- MinTotal :
  - polynomial time algorithm for given fixed total crease width k
  - running time is $O(n^{2+k})$

Summary

Poly-time algorithm under some reasonable assumption?

- MinMax : NP-complete

Computational Complexity (NP-complete?)

- MinTotal :
  - polynomial time algorithm for given fixed total crease width k
  - running time is $O(n^{2+k})$

★ the algorithm indeed runs in $O(2^k n^3)$ (by Yoshio Okamoto)

Fixed Parameter Tractable!!

最新情報

- 折り目を等間隔でないものにした，より一般的なモデルにおける同様の結果が以下の国際会議で発表予定：
  - 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, 2015/02/26-02/28, Dhaka, Bangladesh.
  - 折り目の「厚さ」の定義にいろいろと考えられるけれど，本質的には crease width と同様の結果が得られた.