Folding and Punching Paper

Yasuhiko Asao* Erik D. Demaine† Martin L. Demaine† Hideaki Hosaka‡
Akitoshi Kawamura§ Tomohiro Tachi¶ Kazune Takahashi*

Abstract

We show how to fold a piece of paper and punch one hole so as to produce any desired pattern of holes.

1 Introduction

In the *fold-and-cut problem* introduced at JCDCG'98 [DDL98], we are given a planar straight-line graph drawn on a piece of paper, and the goal is to fold the paper flat so that exactly the vertices and edges of the graph (and no other points of paper) map to a common line. Thus, one cut along that straight line (and unfolding the paper) produces exactly the given pattern of cuts. This problem always has a solution [DO07, BDEH01], though so far the number of folds depends on both the number n of vertices and the ratio r of the largest and smallest distances between nonincident vertices and edges. (A rough estimate on the number of folds is $O(nr)$.)

In the *fold-and-punch problem*, we are given n points drawn on a piece of paper, and the goal is to fold the paper flat so that exactly those points (and no other points of paper) map to a common point. Thus, punching one hole at that point (and unfolding the paper) produces exactly the given pattern of holes. This problem is a natural analog of the fold-and-cut problem where we replace one-dimensional features and target (segments onto a common line) with zero-dimensional features and target (points onto a common point); thus, we also call the problem *zero-dimensional fold and cut.* This problem is also a special case of the *multidimensional fold-and-cut problem* posed in [DO07, after Open Problem 26.32].

Directly applying a fold-and-cut solution to the graph with n vertices and zero edges does not solve the corresponding fold-and-punch problem, because the n points would come to a common line but not a common point. This discrepancy can be fixed by then making $n-1$ one-layer simple folds along perpendicular bisectors between consecutive points (all perpendicular to the common line).

Our goal in this paper is to find more efficient algorithms for the fold-and-punch problem. Indeed, in all four variations described below, we find solutions that depend polynomially in n and only logarithmically or not at all on r (the ratio of the largest and smallest distances between points); see Table 1.

Problem 1 (0-dimensional fold and cut) *Given n points $p_1, p_2, \ldots, p_n$ on a piece of paper, find a flat folding f such that*

$$f(p_1) = f(p_2) = \cdots = f(p_n) \neq f(q) \text{ for all } q \neq p_i.$$

If such a folding exists, what is the order of the number of folds?

We have four variations of this problem based on the following two criteria:

1. Finite paper or infinite paper
2. Allow or forbid crease lines through given points

The second criterion is motivated by the observation that creases passing through given points may lead to a difficulty in the actual punching operation because it has zero tolerance; a small misalignment leads to missing hole or duplicated holes. For example, the fold-and-cut solution places creases passing through the given points.

Theorem 2 *Problem 1 is always solvable in all cases above. The orders of the number of folds (number of folding steps, each of which is composed of either a simple fold or a folding with $O(1)$ creases) and the number of resulting creases in the crease pattern are stated in the following table.*

	Crease Passing		Crease Not Passing	
	Folds	Creases	Folds	Creases
Finite	$O(n)$	$O(n)$	$O(n \log r)$	$O(n^2 r)$
Infinite	$O(n)$	$O(n^2)$	$O(n \log r)$	$O(n^2 r)$

Table 1: Results: Number of folds and resulting creases required in each of the four problem variants.

In the rest of this abstract, we show the sketch of proof of the following two cases: (1) finite paper, allowing crease passing and (2) infinite paper, forbidding crease passing.

2 Finite Paper, Allowing Crease Passing

The proof is by construction. The basic strategy is to align multiple points onto a single horizontal line by folding along horizontal creases and then to add bisectors between consecutive points as follows:

*University of Tokyo, {asao,kazune}@ms.u-tokyo.ac.jp
†MIT, {edemaine,mdemaine}@mit.edu
‡Azabu Junior-High School, h_hosaka314@hotmail.com
§University of Tokyo, kawamura@graco.c.u-tokyo.ac.jp
¶University of Tokyo, tachi@idea.c.u-tokyo.ac.jp

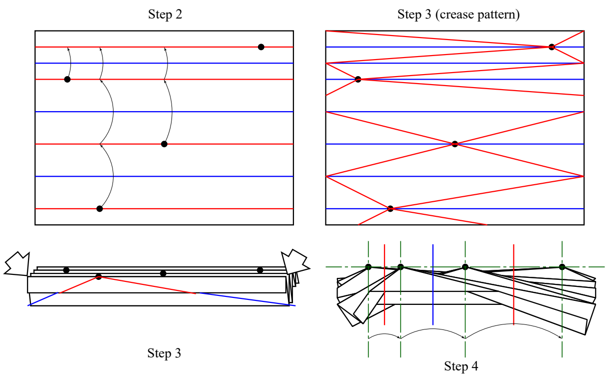

Figure 1: Steps 2–4 to fold given points to a point.

Step 1: Rotate By rotating the paper in the xy-plane, we may assume that the y coordinates $y_1, y_2, \ldots, y_n$ of $p_1, p_2, \ldots, p_n$, respectively, are distinct each other.

Step 2: Horizontally Align Assume $y_1 < y_2 < \cdots y_n$. Then fold the paper along

- mountain creases: lines $y = y_1, y = y_2, \ldots, y = y_n$, and
- valley creases: lines $y = (y_1 + y_2)/2, y = (y_2 + y_3)/2, \ldots, y = (y_{n-1} + y_n)/2$.

As a result, $p_1, p_2, \ldots, p_n$ are on mountain creases and aligned colinearly.

Step 3: Clear Overlaps There exists only one p_i's on each mountain crease. By folding along two slanted lines through p_i, no point except $p_1, \ldots, p_n$ is on the line which $p_1, \ldots, p_n$ are aligned.

Step 4: Vertically Fold Fold along the perpendicular bisectors of adjacent p_i's. This folds p_i's to a single point.

3 Infinite Paper, Forbidding Crease Passing

We introduce *upshifting gadget* to align p_i to a horizontal line while avoiding any part of the paper folded onto p_i. Figure 2 shows an upshifting gadget, which is composed of a pair of twist folds with width d and angle $\theta < 45^\circ$ separating the paper into 6 regions except for the gaps of $3d$ between them. By folding this gadget, these regions get closer to each other. Also, the regions stay singly covered, avoiding other parts of the paper to overlap. If we fix upper-center region to a plane, upper-left(right) region moves to the right (left) by $2d$, bottom-left(right) region moves to upper-right(left) by $2\sqrt{2}d$, and the bottom-center region moves up vertically by $2d$. Here is the detailed steps that replace Steps 2 and 3 of finite crease-passing version.

Step A: Initialize Sort points by its height such that p_1 is the highest point. We draw a horizontal line ℓ passing through p_1. Now consider p_i, the highest point bellow ℓ. i is initially 2.

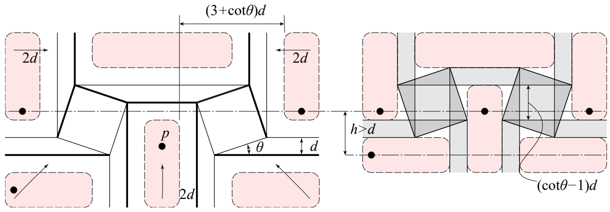

Figure 2: An upshifing gadget that shifts 6 regions painted pink.

Step B: Shrink Let h be the vertical separation between p_i and p_{i+1}. Let 2 the minimum horizontal separation from p_i to other point p_j $(j \neq i)$. Add a horizontal pleat between ℓ and p_i until their distance $2d$ is strictly smaller than $\min(0.5w, 0.5h)$. Here, the number of folds required is at most $O(\log r)$.

Step C: Upshift Insert an upshifting gadget such that $p_0, \ldots, p_{i-1}$ are on either upper-left or upper-right region, p_i is in the bottom-center region, and $p_{i+1} \ldots$ are on either bottom-left or bottom-right region. Fold the gadget to align p_i to ℓ. Increment i and go to Step B until every point is on ℓ.

Combining with the same Steps 1 and 4, we can successfully fold p_i exclusively to a single point.

acknowledgment

We would like to thank Masaki Watanabe for the idea for solving finite paper crease passing case.

References

[BDEH01] Marshall Bern, Erik Demaine, David Eppstein, and Barry Hayes. A disk-packing algorithm for an origami magic trick. In *Origami^3: Proceedings of the 3rd International Meeting of Origami Science, Math, and Education*, pages 17–28, 2001.

[DDL98] Erik D. Demaine, Martin L. Demaine, and Anna Lubiw. Folding and cutting paper. In J. Akiyama, M. Kano, and M. Urabe, editors, *Revised Papers from the Japan Conference on Discrete and Computational Geometry*, volume 1763 of *Lecture Notes in Computer Science*, pages 104–117, 1998.

[DO07] Erik D. Demaine and Joseph O'Rourke. *Geometric Folding Algorithms: Linkages, Origami, Polyhedra.* Cambridge University Press, 2007.