Vol.2010-AL-131 No.11
2010/9/22

折紙における決定不能問題

上　原　隆　平[†1]

近年，計算幾何学の一部で「計算折紙」とよばれる分野が注目を集めている．その分野では，ある意味で折紙を計算のプラットフォームとみなしている．こうしたプラットフォーム上で，計算量理論的に手におえない困難な問題や，多項式時間で解ける問題がいくつか知られている．さてチューリング機械といった標準的な計算モデルにとって，決定不能な問題の存在は，その計算モデルの計算能力の高さを逆説的に示しているといえよう．それならば，計算折紙モデルでは決定不能な問題が存在するだろうか？本稿では，この疑問に対する解答を与える．具体的には，計算折紙モデルのごく自然な決定問題が決定不能であることを示す．

Undecidability of a Simple Origami Problem

Ryuhei Uehara[†1]

Origami has recently attracted much attention as "computational origami". In a sense, origami can be seen as a platform of computation. Both of tractable and intractable results have been obtained on the platform. For a computation model like Turing machine, undecidable problems are a kind of paradoxical evidence of the computational power of the model. Then, is there any undecidable problem on computational origami model? In this paper, we give an affirmative answer. We show that a natural and simple decision problem in an origami computation model is undecidable.

1. Introduction

The idea "computational origami" has recently attracted much attention as theoretical computer science[2]. Since NP-hardness result by Bern and Hayes[1], there are several intractable results from the viewpoint of computational complexity. On the other hand, some origami related software are developed. Among them, TreeMaker, developed by Lang, is well known software for origami design[3]. In this software, it solves several combinatorial optimization problems in a practical time. In this area, origami can be seen as a platform of computation in a sense. We can "compute" some points by folding it using some basic operations on it.

In theoretical computer science, it is known that a computation by a Turing machine is essentially equivalent to recursive function. These natural models are strong enough, and in a sense, this is why their computational power has their limit. For example, consider the following problem, that is well known as the *halting problem*:

Input: A program code P and an input x to P.

Output: Determine whether P will halt with the input x in a finite number of steps.

The halting problem is a simple undecidable problem. That is, there is no program Q solving the halting problem. Since Gödel's incompleteness theorems, such a limit of computation is a paradoxical evidence of the power of a computation system.

Then, how about origami? Is the idea "computational origami" strong enough so that it derives such a paradoxical limit? In this paper, we give an affirmative answer. In a reasonable model of computational origami, we give a natural and simple undecidable problem, that is named *foldability problem*.

Input: An origami with four points p, q, r, and s on it.

Output: Determine whether we can fold two lines ℓ_1 and ℓ_2 such that (1) they are folded by a finite number of operations starting from p, q, r, and (2) they cross at s.

Roughly speaking, the foldability problem asks if we can fold a given point s from just other given three points p, q, r in a finite steps. This is a quite natural problem as origami, but it is, surprisingly, undecidable. We can prove a simpler version of the foldability problem in 1D. That is, the following simpler foldability problem in 1D is still undecidable.

Input: A line segment and four points p, q, r, and s on it.

Output: Determine whether we can fold the point s from the other points p, q, and r in a finite number of operations.

†1 北陸先端科学技術大学院大学
Japan Advanced Institute of Science and Technology

The foldability problem in 2D contains the foldability problem in 1D as a special case. Thus we will show the undecidability for the 1D version in this paper.

2. Computation Model and Undecidability

A 1D *origami* P is a finite line segment of 0 thickness. Without loss of generality, we assume that P has length 1 and put on the interval $[0, 1]$ at first. One real number in $[0, 1]$ is used to represent a point. That is, a point p on a 1D origami is specified by a coordinate. We denote the coordinate of p on P by $P(p)$. We also abuse P to denote each folded state of the origami, and $P(p)$ to denote the coordinate of the point p on the folded origami. We note that each coordinate is a *real number*, that is crucial.

On a 2D origami, we use seven basic operations that consist of Hujita's six axioms and Hatori's additional axiom (see[2) [Chapter 19] for further details). That is, one step in an origami is applying one of seven basic operations, and obtain a new line segment that derives some points by crossing other lines. On a 1D origami, possible operations can be simplified as follows; (1) fix a point $P(p)$ for some point p that already exists on P, and fold some paper layers at once at the point $P(p)$, and (2) unfold some folded paper layers at some point $P(p)$. A folding operation puts some point p onto the other part of the paper, and then we can make a new point q such that $P(p) = P(q)$ on the folded state. This is the only way to increase the number of distinguishable points.

Here we discuss the rule of an operation more precisely. In the problem, we are given four real points p, q, r, and s explicitly on a 1D origami. Without loss of generality, we assume that $P(p) = 0$, $P(q) = 1$, $0 < P(r) < 1$, and $0 < P(s) < 1$. We call s the *goal point* and p, q, r *start points*. We can apply the operations only on the start points and derived points from them. That is, we cannot use s as a handhold of an operation. The goal of the foldability problem is to construct one point r' and a folded state P such that $P(r') = P(s)$ on P. Note that once we have a new real point r', we can check whether $P(r') = P(s)$ with accuracy. (Otherwise, we can also check $P(r') < P(s)$ or $P(r') > P(s)$.) That is, we assume that we can determine if two real points are coincide in general. The real points that can be derived from the start points are said to be *foldable* from the start points.

We first state a theorem for the number of foldable points:

Theorem1 Fix the start points p, q, r such that $P(p) = 0$, $P(q) = 1$ and $0 < P(r) < 1$ on a 1D paper P. Then the number of foldable points is countable.

Proof. Let $S_0 = \{p, q, r\}$ and S_i with $i > 0$ be defined as follows: S_i contains a point t if and only if (1) t is foldable after i folding operations from the start points, (2) $t \notin \cup_{0 \le j < i} S_j$. That is, S_i consists of the points that are folded by exactly i folding operations. Then we can observe that each $|S_i|$ is countable since the number of folded states of the paper P after i folding operations is also countable. Hence each S_i contains countable number of points that implies that the foldable points are countable. ∎

It is well known that a set of real numbers is not countable. Thus, by Theorem 1, we can observe that there exists unfoldable points from given start points. This fact leads us to undecidability:

Theorem2 The foldability problem is undecidable even in the 1D origami model.

Proof. To derive a contradiction, we assume that we have an algorithm A that solves the foldability problem. That is, A always outputs "Yes" or "No" for any points p, q, r and s in a finite time. Since A is an algorithm that can be represented by a programming language on a Turing machine, we can define a function $t_A(p, q, r, s)$ by the number of steps required to output "Yes" or "No" for the input p, q, r, s. By assumption, $t_A(p, q, r, s)$ is finite for any input.

We now fix the start points p, q, and r by $p = 0$, $q = 1$ and, say, $r = 1/\sqrt{2}$. Let T_i be the set of points s such that $T_i = \{s \mid t_A(p, q, r, s) = i\}$. We here prove that $|T_i|$ is countable. Here T_i contains two kinds of points; let Y_i be the set of points s such that A outputs "Yes" for the p, q, r and s, and let N_i be the set of points s such that A outputs "No" for the p, q, r and s. By the definition of the operation, Y_i is countable. That is, A outputs "Yes" because it puts another point s' onto s for some s' and $i' < i$ with $s' \in T_{i'}$. However, N_i might contain infinitely many points with some reason. We prove that such a case cannot occur. If N_i contain infinitely many points, there is an open interval (a, b) with $0 < a < b < 1$ such that all points in (a, b) are in N_i. Then $a' = 0$ and $b' = 1$ are the folded points on the paper with $0 = a' < a < b < b' = 1$. We put a' on b' and make a folded point $c(= 1/2)$ with $|a'c| = |b'c|$. If c is in (a, b), we have a contradiction since c is "Yes" instance. Thus we have either $a' < c < a < b < b'$ or $a' < a < b < c < b'$. If $c < a$, we replace a' by c; otherwise, replace b' by c. Repeating this process finitely many times, we can put the center point c between a' and b' in (a, b). This is a contradiction. Thus N_i can contain finitely many points. Thus $|T_i|$ is countable. Hence $\cup_{0\le j\le i} T_j$ is countable for any fixed integer j. This implies that the size of a set of decidable points by A in a finite time is countable. We let $s_1 < s_2 < s_3 < \ldots$ are the points decidable by A.

Now, by a diagonalization, we can construct a real point s which is not decidable. More precisely, we let (remind that an origami has a unit length)

$$
\begin{aligned}
s_1 &= 0.s_{1,1}s_{1,2}s_{1,3}\ldots \\
s_2 &= 0.s_{2,1}s_{2,2}s_{2,3}\ldots \\
&\ldots \\
s_i &= 0.s_{i,1}s_{i,2}s_{i,3}\ldots \\
&\ldots
\end{aligned}
$$

Then we define $s = 0.s_1s_2s_3\ldots$, where $s_i = s_{i,i} + 1 \pmod{10}$. Then s is a point on the 1D origami, but it does not appear in any T_i.

Hence $t_A(p, q, r, s)$ is not finite for the s. Consequently, the algorithm A does not halt for the input p, q, r, and this s. This is a contradiction that A solves the foldability problem in a finite time. Thus, the foldability problem is undecidable even in 1D origami model. ∎

参 考 文 献

1) M.Bern and B.Hayes. The Complexity of Flat Origami. In *Proc. 7th Ann. ACM-SIAM Symp. on Discrete Algorithms*, pages 175–183. ACM, 1996.
2) E.D. Demaine and J.O'Rourke. *Geometric Folding Algorithms: Linkages, Origami, Polyhedra*. Cambridge University Press, 2007. （邦訳：『幾何的な折りアルゴリズム』，上原隆平訳，近代科学社，2009.）
3) R.J. Lang. *Origami Design Secrets*. A K Peters Ltd., 2003.