Title	多面体と多角形の間の折りと展開に関する研究
Author(s)	鎌田，斗南
Citation	
Issue Date	2023-03
Type	Thesis or Dissertation
Text version	ETD
URL	http://hdl.handle.net/10119/18417
Rights	
Description	Supervisor:上原 隆平，先端科学技術研究科，博士

Doctoral Thesis

Research on Folding and Unfolding between Polygons and Polyhedra

Tonan Kamata

Supervisor: Ryuhei Uehara

Graduate School of Advanced Science and Technology
Japan Advanced Institute of Science and Technology
(Information Science)
2023, 3

Abstract

This paper aims to clarify the folding/unfolding relation between polygons and polyhedra. A polyhedron Q is called unfoldable into a polygon P if we obtain P by cutting a certain set of line segments (not limited to edges) on the surface of Q. Inversely, a polygon P is called foldable into a polyhedron Q if Q is unfoldable into P.

The first part of the thesis is a folding problem that inquires whether a polygon P is foldable into a polyhedron Q for given P and Q. An efficient algorithm for this problem when Q is a box was recently developed. We extend this idea to a class of convex polyhedra. We develop two algorithms for the problem. The first algorithm solves the folding problem for a certain class of convex polyhedra, with a unit length and a unit angle, except for tetramonohedra. The second algorithm handles the exceptional case for the class of tetramonohedra. Combining these algorithms, we can conclude that the folding problem can be solved in pseudo-polynomial time when Q is a polyhedron in a certain class of convex polyhedra, which includes Platonic solids.

The second part of this thesis is a reconfiguration problem on refolding. We show that any pair of polyhedra in several classes of polyhedra is joined by a sequence of $O(1)$ refolding steps, where each refolding step unfolds the current polyhedron into a polygon that is foldable into the next polyhedron. In other words, a polyhedron is refoldable into another polyhedron if they share a common unfolding. Specifically, we prove that (1) any two tetramonohedra are refoldable into each other, (2) any doubly covered triangle is refoldable into a tetramonohedron, (3) any tetrahedron has a 3-step refolding sequence to a tetramonohedron, (4) any (augmented) regular prismatoid and doubly covered regular polygon are refoldable into tetramonohedra, and (5) the regular dodecahedron has a 4-step refolding sequence to a tetramonohedron. In particular, we obtain a ≤ 6-step refolding sequence between any pair of Platonic solids, applying (5) for the dodecahedron and (1) and/or (2) for all other Platonic solids.

The third part of this thesis is about the nonexistence of common unfoldings. We show that the existence of common unfoldings can be reduced to the existence of standard-form common unfoldings under a certain condition. We also develop an algorithm that checks the existence of standard-form common unfoldings, and we implement it on some specific polyhedral class. We obtain the fact that there is no common unfolding with k vertices within $k < 300$ between any strongly-independent and algebraic doubly covered triangles.

Keywords: Computational geometry, Computational origami, Unfolding of polyhedra, Common unfolding, Refolding, Reconfiguration problem.

Acknowledgments

First, I am very grateful to my supervisor, Prof. Ryuhei Uehara. He has always welcomed discussions with me and suggested directions through his high level of expertise. He provided me invaluable opportunities and environments for this research.

I am also grateful to my collaborators. Prof. Erik D. Demaine, in particular, has accepted me to his laboratory at the Massachusetts Institute of Technology for over a month. The result of Chapter 5 deeply leads to this visit.

I thank the past and present members of our laboratory, who discussed with me and gave me many notices. Especially, Ms. Tomoko Taniguchi always carefully checked my English texts during the research and gave me precise comments.

I would like to express my appreciation to my parents. They have clearly and serenely bucked me up.

Last but not least, I am grateful to my partner, who always makes a lot of effort to understand my research, supports me warmly, and broadens my horizons.

Contents

Chapter 1

Introduction

1.1 Background

To make some objects from paper, we draw its development view on the paper, punch it out, and construct it by gluing its boundary. If we can obtain a connected and non-overlap development view, it is useful to make or figure out the object. The origin of this idea is back to the German painter Albrecht Dürer. In 1525, he published his masterwork on geometry [10], in which he presented each polyhedron by drawing the development of the surface of the polyhedron to a planar layout without overlapping when cutting along its edges. This representation is known under different names; net, development, or unfolding and has been mainly studied in discrete geometry. In this paper, we use the terms unfolding or edge-unfolding depending on whether we allow cutting lines to cross the faces or restrict cutting lines on the edges of the polyhedron. We do not care whether unfoldings overlap unless otherwise noted.

One of the most significant results of unfoldings is a characterization of the unfoldings of tetramonohedra by Akiyama et al. [1, 2]. Tetramonohedra is the class of tetrahedra with four congruent faces. In [1, 2], Akiyama et al. characterize the unfoldings of tetramonohedra by using tiling theory and show that a polygon is an unfolding of a tetramonohedron if and only if P satisfies a condition called Conway criterion (see Section 2.4 for details). This result illustrates the potential of the research on the relationship between polyhedra and these unfoldings.

Nowadays, unfoldings have been well studied in computational geometry, forming a research field called computational origami. Demaine and O'Rourke systematically summarize results and open problems in the field in [9].

In [9], there are two open problems that are simple but quite difficult. The first is "Does every convex polyhedron have a non-overlap edge unfolding? (Open Problem 21.1)". It is a natural extension of Dürer's result and has been studied from negative and positive expectations with some corroborative results (see [9] for details). The second is "Is any Platonic solid refoldable into a different Platonic solid? (Open Problem 25.6)". A polyhedron Q is refoldable into a polyhedron Q' if we can unfold Q into a polygon and fold the polygon into Q'. In other words, Q is refoldable into Q' if they share a common unfolding. This question is also natural but open.

To get a foothold of Open Problem 21.1, the authors of [9] propose a framework called "bipartite space of foldings and unfoldings" (hereinafter called "FUB-space"). FUB-space has two vertical sets; all convex polyhedra and all polygons. A pair of a polyhedron Q and a polygon P is connected by an edge if P can be folded to Q, or equivalently Q can be unfolded to P (see Figure 1.1).

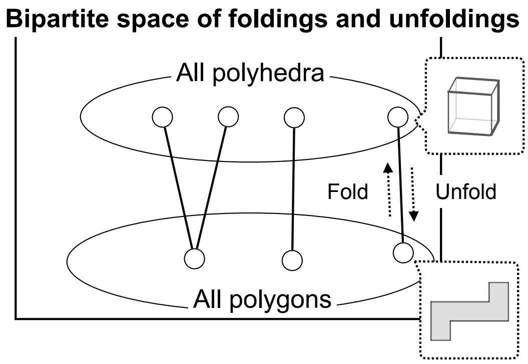

Figure 1.1: Bipartite space of foldings and unfoldings

FUB-space has been studied mainly under the restriction that we cut only the edges of polyhedra. Specific results can be found in [5, 6, 12]. This is because the polygons that can be developed from a single polyhedron would exist continuously (see Figure 1.2), which is difficult to handle especially on a computer. However, where the edges has no clear meaning for the surface structure of a polyhedron. Thus, dealing only with edge unfolding is far from enough to understand the relationship between polyhedra and polygons.

1.2 Purpose of the research

This study considers how to deal discretely with the relationship between polyhedra and unfoldings without restrictions on the way to unfold and fold

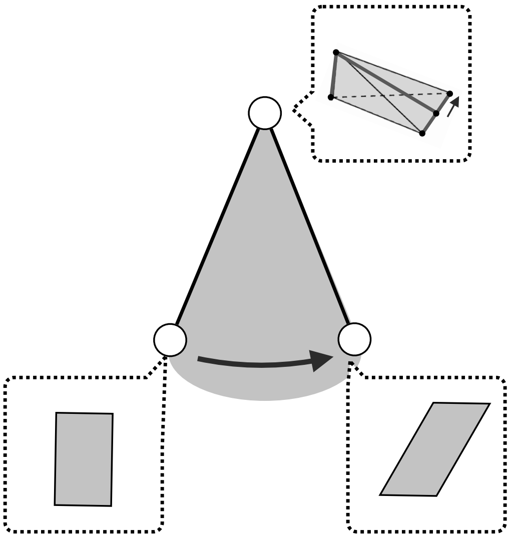

Figure 1.2: An example of continuous transformation of an unfolding by moving the cut line

them. We focus on discrete structures that appear in the folding and unfolding operations through the extension of established problems and the development of new ones.

In Chapter 3, we consider a folding problem. This problem asks whether a polygon P and a polyhedron Q are connected by an edge in FUB-graph for fixed P and Q (see Figure 1.3).

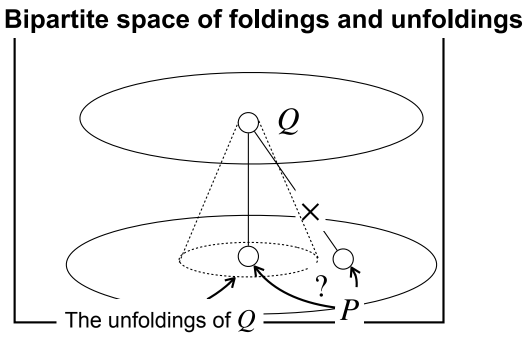

Figure 1.3: Folding Problem

What we show is that ways to fold a polygon into a polyhedron exist discretely except in special cases and that the number of ways to fold can be bounded by a pseudo-polynomial of geometric parameters under certain conditions. The folding problem was investigated where Q is a box. Some special cases were investigated in [11] and [15], and the problem for a box Q was generally solved in [14]. In this research, we develop a more efficient algorithm that can be applied more generally. What we focus on is that an unfolding of a polyhedron Q has some “trace” of Q on its boundary if Q is not a tetramonohedron (Lemma 14). Using this fact, we develop an algorithm that solves the folding problem where Q is not a tetramonohedron but a commensurate convex polyhedron; a polyhedron with a unit length and angle. To compensate for this result, we also develop an algorithm in the case where Q is a tetramonohedron by using the results of [1, 2].

In Chapter 4, we consider a reconfiguration problem on refolding. This problem asks whether a pair of polyhedra Q and Q' is connected by a finite length path in FUB-graph for fixed Q and Q' (see Figure 1.4).

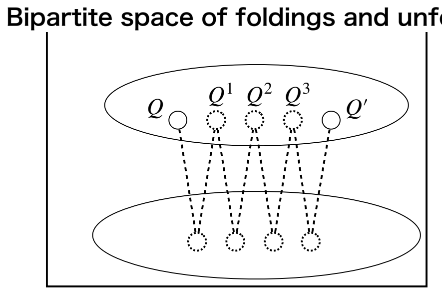

Figure 1.4: Reconfiguration Problem on Refolding

What we show is that there exists a continuous set of polyhedra connected to each other by finite lengths and that some polyhedra can be reduced to the class by choosing a path from discrete choices. Specifically, we consider a sequence of convex polyhedra $Q = Q_0, Q_1, \ldots, Q_k = Q'$ where Q_{i-1} is refoldable into Q_i for each $i \in \{1, \ldots, k\}$ for a given pair of polyhedra Q and Q'. As we mentioned above, a polyhedron Q is refoldable into a polyhedron Q' if Q can be unfolded to a polygon that can be folded to Q'. We consider the reconfiguration by a refolding sequence between convex polyhedra. A reconfiguration problem is one of the important topics in theoretical computer science. It is well studied from wide perspectives, e.g., (im-)possibility of reconfiguration, the time complexity of an algorithm to find a path and the shortestness of a path. First, we show that any two tetramonohedra are 1-step refoldable into each other. Next, we show that any polyhedra with ≤ 4 vertices (they exist continuously) can be reduced to tetramonohedra by ≤ 4 steps. Finally, we show that a polyhedron in several polyhedral classes is connected into a tetramonohedron by a $O(1)$ refolding sequence. All Platonic solids are included in these results, and we obtain a ≤ 6-step refolding sequence between any pair of Platonic solids via tetramonohedra.

In Chapter 5, we discuss the non-existence of common unfoldings. This problem asks whether a pair of polyhedra Q and Q' is not connected by a 2-step path in FUB-graph for fixed Q and Q' (see Figure 1.5).

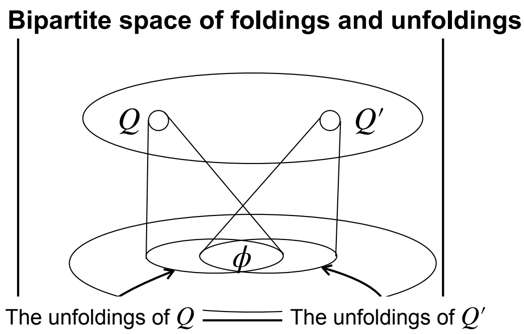

Figure 1.5: Non-existence of Common Unfolding Problem

What we show is that the existence of common unfolding can be reduced to a discrete structure called standard-forms under a certain condition. First, we focus on the relation between points on the boundary of a common unfolding. If two polyhedra Q and Q' have a common unfolding P, P can be folded into Q and Q'. In general, a point p in $\partial(P)$ is attached to different points when folded one way or the other. We show that this relation induces a tree structure on the boundary of the polygon called spreading trees. We define standard forms of common unfoldings using the spreading trees. We investigate an algorithm to enumerate the possible standard forms. We implement this algorithm in a class of restricted doubly covered triangles and check that no k-gon is a common unfolding within $k < 300$.

1.3 Relationship among the results

This research aims to characterize the relationship between polyhedra and their unfoldings. In particular, we aim to find the criteria that two polyhedra have a common unfolding or not. To reach this goal, we clarify the positive and negative aspects of the existence of common unfolding while adjusting the class of polyhedra on each result.

In Chapter 3, we introduce the class called "commensurate convex". This class consists of widely many polyhedra, which include most of the polyhedra we normally deal with. In fact, all of Platonic solids and Archimedean solids are commensurate convex. We show that, for widely many polyhedra, an unfolding of a polyhedron has a "trace" of the original polyhedron, and it can be used for an efficient recognition algorithm for unfoldings.

The existence of the trace makes it difficult to make a common unfolding between a given pair of polyhedra. In Chapter 4, we introduce a new idea of “multistep refolds” and show that we can create and erase the traces by repeating the several refoldings among a certain polyhedra class with few vertices or some symmetry.

Even though the above two results imply the difficulty of making a common unfolding, it is difficult to show that there is no common unfolding between a pair of polyhedra. In Chapter 5, we define the polyhedral classes called “strongly independent” and “algebraic”. These are a technical and tight class, with very few conformable polyhedra. Actually, when two polyhedra are commensurate, each is not algebraic, and the pair is not strongly independent. Under these restrictions, we show the nonexistence of a common unfolding with the condition that unfoldings have few vertices.

Chapter 2

Preliminaries

2.1 Polygons and polyhedra

A ***polygonal line*** P is defined by a sequence $(p_0, p_1, \ldots, p_n)$ of distinct points in $\mathbb{R}^2$. A point p_i is called ***vertex*** of P, and $V(P)$ denotes the set of those. We call the open interval e_i between p_i and p_{i+1} by an ***edge***, and $E(P)$ denotes the set of them. We denote the set of all vertices and the points on the edges by $\partial(P)$. A ***polygon*** is a polygonal line that satisfies $p_0 = p_n$. A polygon is called ***simple*** if any edge has no intersection with other edges or vertices except at its endpoints. A simple polygon divides $\mathbb{R}^2$ into two regions: a finite one and an infinite one. Let $\angle(p_i)$ be the angles at p_i on the interior side.

We define a ***(convex) polyhedron*** Q as the surface of a convex region of $\mathbb{R}^3$ that is bounded by finitely many polygons $F(Q) = \{f_0, f_2, \ldots, f_{m-1}\}$, which are called ***faces***. For any $f_i, f_j \in F(Q)$, $\partial(f_i) \cap \partial(f_j)$ is $\emptyset$ or a vertex or an edge. The vertices shared faces are called ***vertices*** of Q, denoted by $V(Q)$. Similarly, the edges shared two faces are called ***edges*** of Q and denoted by $E(Q)$. We denote the set of the points included in faces, edges, or vertices by $\partial(Q)$ and call it by a ***surface*** of Q. Every edge e joins two adjacent faces with some angle. The angle on the interior side is called a ***dihedral angle*** at e.

For $q \in \partial(Q)$, a ***curvature*** $\kappa(q)$ is the angle defined by the value $2\pi - \sigma(q)$, and a ***co-curvature*** at q is the angle $\sigma(q)$, where $\sigma(q)$ is the total summation of the angles on the faces of Q adjacent to q. If $q \notin V(Q)$, $\sigma(q) = 2\pi$ and $\kappa(q) = 0$. Any convex polyhedron has positive curvatures at the vertices.

We will use the following theorem:

Theorem 1 (Descartes Theorem (Discretized Gauss-Bonnet Theorem))

For any convex polyhedron Q*,*

$$\sum_{v \in V(Q)} \kappa(v) = 4\pi.$$

See [9, Sec. 21.3] for the details.

2.2 Polyhedral classes

Here, we introduce polyhedral classes that are commonly used in this thesis.

2.2.1 Classification by smooth vertices

If a vertex v of a polyhedron satisfies $\kappa(v) = \sigma(v) = \pi$, we call v ***smooth***. We define $\boldsymbol{\Pi_k}$ as the class of polyhedra with exactly k smooth vertices. By Theorem 1, the number of smooth vertices of a convex polyhedron is at most 4. Therefore, the classes $\Pi_0, \Pi_1, \Pi_2, \Pi_3$, and Π_4 give us a partition of all convex polyhedra.

2.2.2 Doubly covered polygon

A doubly covered polygon is a polyhedron made by gluing the corresponding edges of two copies of a convex polygon; see Figure 2.1. It can be regarded as a kind of polyhedron whose volume is zero.

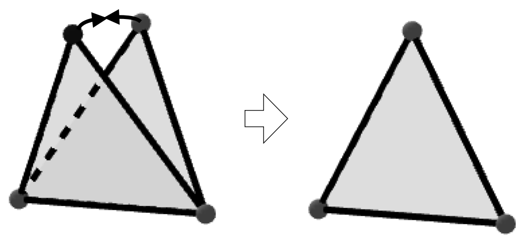

Figure 2.1: Doubly covered triangle

2.2.3 Platonic solids

Platonic solids form a class of polyhedra whose all faces are congruent regular polygons, and all vertices are congruent. It is known that the set of Platonic solids consists by five polyhedra; regular tetrahedron, regular hexahedron, regular octahedron, regular dodecahedron, and regular icosahedron.

2.3 Folding and unfolding

For a simple polygon P, ***crease lines*** CL are defined by a set of non-crossing lines on $F(P)$ whose both endpoints are on $\partial(P)$. For a given CL, we can obtain a folded 3-dimensional surface by assigning a dihedral angle for each crease line and folding them along the dihedral angles. If the resulting surface makes a polyhedron Q without overlap or gaps, we say that we can ***fold*** P into Q. Inversely, we can unfold a polyhedron Q into a (not necessarily simple) polygon by cutting the surface. We call the resulting polygon ***unfolding***.

The following theorem, shown in [9, Sec. 22.1.3], is important in considering unfoldings.

Theorem 2 *When we unfold a convex polyhedron Q into a polygon, the cutting line segments form a tree structure on the surface of Q and span all vertices of Q.*

This theorem holds because the resulting polygon is divided into two pieces if the cutting lines contain a circle and because a vertex that the cutting tree does not pass through is not flattened. From this fact, we call the cutting line segments ***cutting tree***, and denote by T. We assume that any cut ends at a point with a less than 2π curvature. Otherwise, because Q is convex, it makes a redundant cut on P, which can be eliminated (the proof can be found in [13, Theorem 3]). For a point $q \in T$, let $deg(q)$ be the number of the line segments of T incident to q.

When a polyhedron Q is unfolded into a polygon P with a cutting tree T, the points on $\partial(P)$ correspond to points on T. We call this correspondence ***folding map*** and write it by $f_T : \partial(P) \to T \subset \partial(Q)$.

For $q \in T$, the inverse image $f_T^{-1}(q)$ is the set of the points gathering at q when we fold P. The following holds.

Observation 3

$$\sum_{p \in f_T^{-1}(q)} \angle(p) = \sigma(q)$$

Especially, $\angle(p) = \sigma(q)$ holds if $deg(q) = 1$.

The following theorem is useful for constructing a polyhedron from a polygon.

Theorem 4 (Alexandrov's Theorem [3, 9]) *If we fold a polygon P in a way that satisfies the following three **Alexandrov's conditions**, then there is a unique convex polyhedron Q realized by the folding.*

1. *Every point on $\partial(P)$ is used in the gluing.*

2. *At any glued point on the surface, the co-curvature is at most 2π.*

3. *The obtained surface is homeomorphic to a sphere.*

This theorem gives no information about the specific shape of the polyhedron but guarantees the existence of the polyhedron when we glue the boundary of a polygon.

2.4 Tetramonohedron

A ***tetramonohedron*** is a tetrahedron that consists of four congruent acute triangles. The following lemma, shown in [4, p. 97], characterizes tetramonohedra.

Lemma 5 *A polyhedron is a tetramonohedron if and only if it is in Π_4.*

A polygon P is a ***Conway tile*** if it has six points, A, B, C, D, E, and F in $\partial(P)$ in a counterclockwise direction that satisfy the following conditions (see Figure 2.2). Let $[p, q]$ denote the part of the boundary $\partial(P)$ starting from p to q in a counterclockwise direction for two points p and q.

- $[A, B]$ can be moved to $[D, E]$ through translation τ with $\tau(A) = E$ and $\tau(B) = D$.

- each of $[B, C]$, $[C, D]$, $[E, F]$, and $[F, A]$ has rotational symmetry with respect to its midpoint.

- at least three of these six points are distinct.

Akiyama et al. [2, 1] characterize unfoldings of tetramonohedra by the following theorem.

Theorem 6 *A polygon P is an unfolding of a tetramonohedron if and only if P is a Conway tile.*

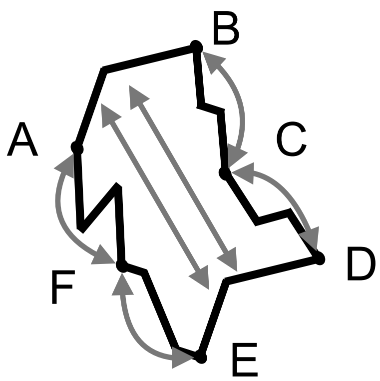

Figure 2.2: Conway tile

Chapter 3

Folding problem

This chapter focuses on a folding problem that asks whether a polygon P can be folded into a polyhedron Q for given P and Q. More precisely, P can fold onto Q if and only if there is a set of crease lines on P such that we obtain Q by folding P along the crease lines and gluing P without overlap or gaps on Q. We assume that P and Q have the same (surface) area without loss of generality as a trivial necessary condition.

3.1 Settings

We assume the real RAM model for the computations; each coordinate is an exact real number, and the running time is measured by the number of mathematical operations. See [7] for the details of the basic operations, which can be applied in $O(1)$ time for this data structure.

To describe a polyhedron Q, we use the doubly-connected edge list in [7] with some trivial extensions.

We introduce a new class of convex polyhedra: a convex polyhedron Q is a ***commensurate convex polyhedron*** with respect to $\ell_{\min}$ and $\theta_{\min}$ if Q has a unit length $\ell_{\min}$ and a unit angle $\theta_{\min}$ in the sense that each edge is a multiple of $\ell_{\min}$ and each angle of a face of Q is a multiple of $\theta_{\min}$. (That is, $\ell_{\min}$ and $\theta_{\min}$ are the great common divisors of the lengths of the edges and the angles of the faces on Q, respectively.) We define a constant c by the minimum integer for which $c\theta_{\min}$ is a multiple of π. Here, we provide proof of the existence of c.

Observation 7 *We assume that a convex polyhedron Q is a commensurate convex polyhedron with respect to $\ell_{\min}$ and $\theta_{\min}$. There then exists an integer c such that $c\theta_{\min}$ is a multiple of π.*

Proof. Because Q is a convex polyhedron, each face is a convex polygon. Let f be a face of Q. Then, f is a convex k-gon for some $k > 2$, and the summation of angles of f is $(k-2)\pi$. Based on the assumption, this $(k-2)\pi$ is a multiple of $\theta_{\min}$. Therefore, an integer c exists such that $c\theta_{\min}$ is a multiple of π. □

We define ***diameter*** D_P and ***perimeter*** L_P for a given polygon P as follows:

$$\begin{aligned} D_P &= \max_{p,p' \text{on } \partial(P)} |pp'| \\ L_P &= \sum_{0 \le i < n} |p_i p_{i+1}|, \end{aligned}$$

where $|pq|$ is the distance between two points p and q. When polygon P is clear, the subscript P is omitted. We also denote by $\ell_{\max}$ the length of the longest edge of P defined by $\max_{0 \le i < n} |p_i p_{i+1}|$. We observe that $\ell_{\max} \le D$ and $2D < L$ for any simple polygon P.

3.2 Results and framework

Here, we state our results.

Theorem 8 *Let P be a polygon with n vertices and Q be a commensurate convex polyhedron with m vertices that is not a tetramonohedron.*

There is a pseudo-polynomial time algorithm that solves the folding problem in $O\left(n^2 m^3 \left(dL + n + \frac{2\pi}{\theta_{\min}} n\right)^c\right)$.

Theorem 9 *Let P be a polygon with n vertices and Q be a tetramonohedron.*

There is a pseudo-polynomial algorithm that solves the folding problem in $O((L+n)^2 n^2)$ *time.*

These algorithms are commonly based on the outline of Algorithm 1.

To explain the framework, we focus on what happens when a polygon P can be folded into a polyhedron Q. Let T be the cutting tree T and f_T be the folding map. We take a point p on $\partial(P)$ such that $f_T(p)$ is a vertex q of Q. Let ϵ be a smallness value and p_ϵ be the point that is moved ϵ from p counter-clock-wisely on $\partial(P)$. The point $f_T(p_\epsilon)$ should be on a face f around q on Q. We set Q onto the xy-plane such that q meets the origin and its edge of f meets the x-axis (see Figure 3.1). We also set P on the xy-plane and fix p to the origin. By adjusting the direction of P, we can match p_ϵ and $f_T(p_\epsilon)$

Algorithm 1: Outline of our folding algorithm

Input : A polygon P and a polyhedron Q
Output: All ways of folding P onto Q (if one exists)
Enumerate the initial positions of P and Q by ***EnumPosi***.
foreach ***combination of initial positions of P and Q*** **do**
  Set Q and P on the xy plane according to the initial positions.
  Wrap the surface of Q by P by ***Stamp***.
  if ***all vertices of Q are touched by $\partial(P)$*** **then**
    Check whether the wrapping is without overlap or gaps by ***GlueCheck***.
    if ***there is no overlap or gap*** **then**
      Output: the way of folding

(see Figure 3.2). After the positions of P and Q are determined, we would wrap P onto the surface of Q: we fold P along the edge with its dihedral angle whenever P overhangs an edge of Q. As a result, we obtain the folding way such that $\partial(P)$ is glued doubly covered and makes T. Inversely, if P can be folded into Q, at least one pair of positions of P and Q must achieve the above wrapping.

Our algorithms use this process inversely: enumerating the possible ways to locate P and Q on the xy plane, wrapping P onto the surface Q, and checking whether $\partial(P)$ is glued successfully.

The enumerating process is summarized as ***EnumPosi***, given in Section 3.5. First, we choose a face f of Q and a vertex q of f and fix Q on the xy plane such that q is on the origin, the edge e injecting to q counter-clockwisely is on the x axis, and f is on the xy plane. To fix a position of P on the xy plane, we choose a pair (p, p') of the points in $\partial(P)$ and decide the coordinate $((p_x, p_y), (p'_x, p'_y))$. Therefore, what we enumerate in this process is the possible combinations of $f \in F(Q)$, $q \in V(Q)$, $p, p' \in \partial(P)$, and $(p_x, p_y), (p'_x, p'_y) \in \mathbb{R}^2$. We show that we can bound the number of the possible combinations discretely although both $\partial(P)$ and $\mathbb{R}^2$ are uncountable sets in Lemma 14 and Theorem 15.

Here we note that the number of ways of folding P onto Q would be infinite in a special case that Q is a tetramonohedron and the form of the cutting tree is an "H-shape". However, we can also handle this case because either there is one infinite series such that every folding way is feasible or no one is feasible. We give the details in Section A.

After ***EnumPosi***, we try to wrap Q by P for each initial position. This process is done by an algorithm called ***Stamp***. We give the details in Section

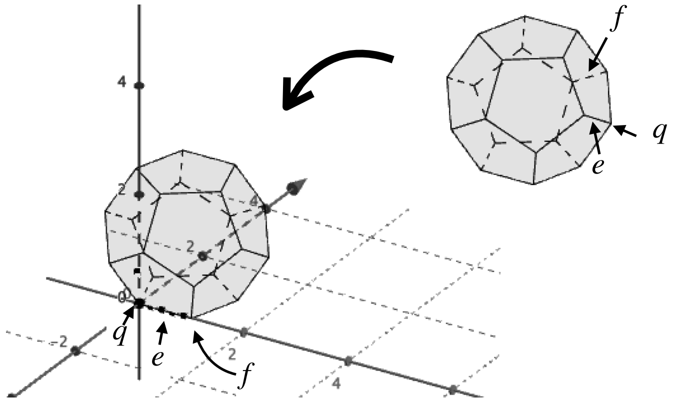

Figure 3.1: The way to fix a position of Q

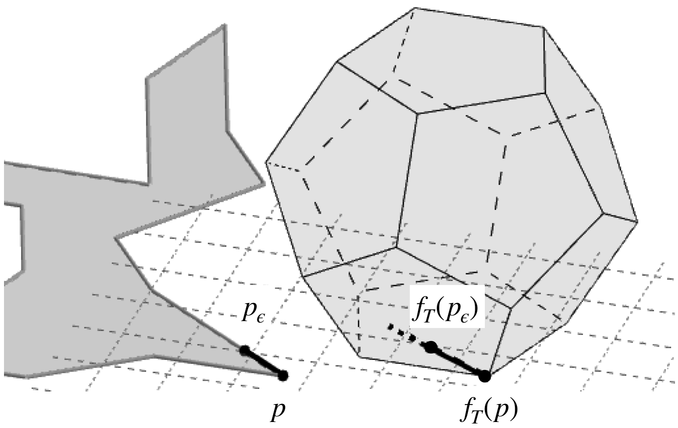

Figure 3.2: Correspondence of (p, p_ϵ) and $(f_T(p), f_T(p_\epsilon))$

3.3. After this process, if a vertex of Q is not touched by $\partial(P)$, the combination of the initial positions is rejected because it does not give a cutting tree by Theorem 2.

Finally, we check whether the folded boundary of P makes overlaps or gaps. This process is done by an algorithm called ***GlueCheck***. We give the details in Section 3.4.

3.3 Stamping

We first give the details of ***Stamp***.

We note that each point inside of a face f_k of Q has a unique local coordinate; a point on an edge of f_k except its endpoints has two local coordinates $(f_k; x, y)$ and $(f_{k'}; x', y')$, where $f_{k'}$ is the face sharing the edge; and each vertex of the polygon f_k has d local coordinates, where d is the number of the faces sharing the vertex on Q.

We assume that each initial position of P and Q is fixed; a vertex q of Q, a vertex f of Q, a pair p, p' of $\partial(P)$, and a coordinate $(p_x, p_y), (p'_x, p'_y)$ of p, p' are determined, and P and Q are settled according to them. As we see later in Section 3.5, we can assume that $f_T(p) = q$ and $(p_x, p_y) = (0, 0)$. By re-labeling, we assume $q = q_0$, $p_0 = p$, $f = f_0$, and $p_0, p_1, \ldots, p_n$ are lined up counter-clock-wisely. The local coordinate of q_0 on f_0 is $(f_0; 0, 0)$.

Next, we trace $\partial(P)$ from p_0 and find the first point c intersecting with the boundary of f_0. While tracing between p_0 to c, we record the local coordinates of the vertices $p_1, p_2, \ldots$ at f_0. If any point that is moved a little from p_0 counter-clock-wisely goes outside of f_0, then let e be the edge that incidents to p_0 counter-clock-wisely on f_0. Otherwise, we can obtain the intersection c. Here, there are two possible cases depending on whether c is a vertex of f_0 or not. If c is not a vertex, we also record its local coordinates and add it to the sequence of the vertices of P as a vertex whose interior angle is π. Also, if c is on an edge of f_0, let e be the edge. Otherwise, if c is a vertex, let e be the edge that incidents to c clock-wisely on f_0. Let f_1 be the face that shares e with f_0.

Next, we rotate Q on e such that f_1 becomes the base face. We trace $\partial(P)$ again and repeat the same process until we return to p_0.

Using an array of the elements of $V(Q)$, we record whether each vertex is touched by $\partial(P)$ during the stamping process. After stamping, we check whether all vertices are touched by tracing the array.

As a result, we obtain a peace-wised P by the intersections of $\partial(P)$ and edges of Q. We denote it by $P' = (p'_0, p'_1, \ldots, p'_{n'-1}, p'_{n'} = p'_0)$. For each of the points of P', a local coordinate of a face of Q is defined.

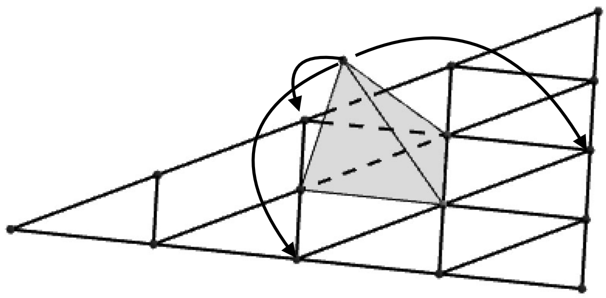

Figure 3.3: A triangle lattice is made by stamping a tetramonohedron.

We note that the faces made by stamping form a triangle lattice during stamping when Q is a tetramonohedron (Figure 3.3). The details are given in [1].

To estimate the time complexity of the process, we give the following lemma.

Lemma 10 *If P is unfolded into Q that is a commensurate convex polyhedron, an edge e goes through at most $d|e|+1$ faces on Q where $d = \frac{c}{(1+c\sin\theta_{\min})\ell_{\min}}$.*

When Q is a tetramonohedron, the upper bound is $3|e|+1$.

Proof. Let e be an edge of length $|e|$ of P, which penetrates faces $f_0, f_1, \ldots,$ of Q when Q is rolling on e. We note that each face f is a convex polygon because Q is a convex polyhedron. When the edge e intersects a face f, the edge e enters a point p on an edge e_Q of f and goes out at another point p' on an edge e'_Q of f.

When e_Q and e'_Q do not share a vertex of f, $|pp'| > (\ell_{\min}\sin\theta_{\min})$. Therefore, the face f consumes $(\ell_{\min}\sin\theta_{\min})$ from e.

On the other hand, we cannot bind it when e_Q and e'_Q are consecutive on f. Namely, the edge e penetrates many faces when e passes through points close to a vertex of Q shared by many faces (Figure 3.4). However, in this case, consecutive c faces consume at least length $\ell_{\min}$ from e, where c is the minimum integer for which $c\theta_{\min}$ is a multiple of π.

Therefore, the edge e of length ℓ can go through at most $\left(\frac{|e|}{(\ell_{\min}\sin\theta_{\min}+\ell_{\min}/c} + 1\right)$ faces, which is simplified to $d|e|+1$, where $d = \frac{c}{(1+c\sin\theta_{\min})\ell_{\min}}$, in total.

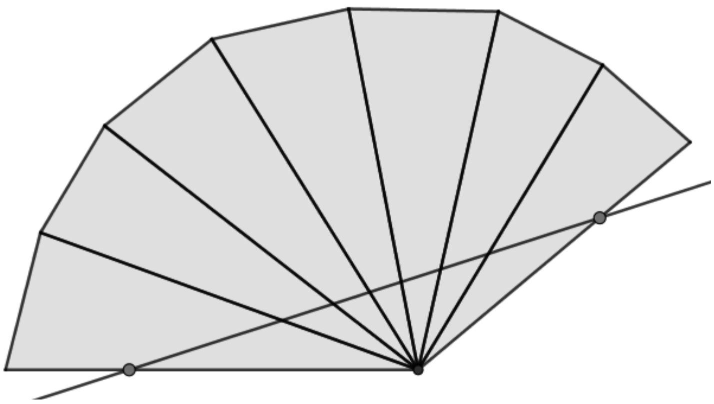

Figure 3.4: An edge e_p may penetrate many different faces.

When Q is a tetramonohedron, the number of penetrates faces sharing a vertex is at most three because triangle faces make a triangle lattice during the stamping of a tetramonohedron. Therefore, an edge e of length $|e|$ of P goes through at most $3|e|+1$ faces. □

Theorem 11 *Let Q be a commensurate convex polyhedron with respect to $\ell_{\min}$ and $\theta_{\min}$. The number of rollings in the stamping process is $(dL+n+\frac{2\pi}{\theta_{\min}}n)$, where $d=\frac{c}{(1+c\sin\theta_{\min})\ell_{\min}}$. Moreover, the stamping process can be done in $O\left((dL+n+\frac{2\pi}{\theta_{\min}}n)m\right)$ time.*

When Q is a tetramonohedron, the number of rollings is $O(L+n)$, and the process can be done in $O(L+n)$ time.

Proof. The stamping process is applied along $\partial(P)$ by rolling Q. Therefore, we can use Lemma 10 on each edge e of P. To cover each edge e of P, we need $(d|e|+1)$ rollings. Moreover, there are extra rollings on each vertex P that do not cover the edges. When a vertex q of Q is placed on a vertex p of P, Q may turn around the vertex p by rolling to cover the next edge of P. Because the angle at p can be close to 2π, the number of turns is bounded above by $2\pi/\theta_{\min}$. Thus, the total number of rollings can be bounded by $dL+n+\frac{2\pi}{\theta_{\min}}n$ because $\sum_{i=0}^{n}(d|p_ip_{i+1}|+1)=dL+n$.

In each rolling step, we compute the intersection of a face f of Q and $\partial(P)$, and it takes $O(m)$ time to traverse the vertices in f. Therefore, we can do stamping process in $O\left((dL+n+\frac{2\pi}{\theta_{\min}}n)m\right)$.

For the case that Q is a tetramonohedron, we can show the claim by Lemma 10. We note that m is a constant number in this case. □

3.4 Check of gluing

During this phase, the algorithm checks if P can fold onto Q by folding along the crease lines given in ***Stamp***. That is, the algorithm checks whether the surface of Q is made by gluing P without overlap or gaps.

This can be achieved by checking the new polygon $P' = (p'_0, p'_1, \ldots, p'_{n'-1}, p'_{n'} = p'_0)$, which is constructed using the stamping. Based on Theorem 2, when P is an unfolding of Q, the set T of cut lines on Q forms a spanning tree. Therefore, each line segment ℓ in T appears twice as ℓ' and ℓ'' on P' (except for its endpoints), and the pairs of ℓ' and ℓ'' form a nested structure on P'. The algorithm can find each pair of line segments in the polygon $P' = (p'_0, p'_1, \ldots, p'_{n'-1}, p'_{n'} = p'_0)$ by using their local coordinates.

We maintain P' by a doubly linked list; each item corresponds to p'_i which stores p'_{i-1}, p'_{i+1}, $\left|p'_i p'_{i-1}\right|$, $|\ell'_i| = \left|p'_i p'_{i+1}\right|$, and $\angle(p'_i) = \angle p'_{i-1} p'_i p'_{i+1}$.

We also inherit the set S of gluing points in P' from the stamping step such that each gluing point p has angle $\sigma(q)$ at the point on $\partial(P)'$ that corresponds to a vertex q of Q. Intuitively, each gluing point should be zipped up from the point to fold onto Q. In other words, each gluing point corresponds to a leaf of the spanning tree T of the vertices of Q to cut and unfold into P.

Therefore, our gluing starts from any gluing point. We first pick up arbitrary gluing point p'_i in S and glue two line segments ℓ'_{i-1} and ℓ'_i from p'_i. We then have two cases. The first case is $\left|\ell'_{i-1}\right| \neq |\ell'_i|$. Without loss of generality, we assume that $\left|\ell'_{i-1}\right| < |\ell'_i|$. In this case, we glue up to p'_{i-1} from p'_i. That is, we remove p'_i from S and P', replace p'_i in p'_{i-1} by p'_{i+1} with length $|\ell'_i| - \left|\ell'_{i-1}\right|$, and replace p'_i in p'_{i+1} by p'_{i-1} with length $|\ell'_i| - \left|\ell'_{i-1}\right|$. The angle $\angle(p'_{i-1})$ at p'_{i-1} is replaced by $\angle(p'_{i-1}) + \pi$.

The second case is $\left|\ell'_{i-1}\right| = |\ell'_i|$. In this case, we glue ℓ'_{i-1} and ℓ'_i completely, remove p'_i from S and P', and merge p'_{i-1} and p'_{i+1} into a new vertex p' on P' such that $\left|p'_{i-2} p'\right| = \left|p'_{i-2} p'_{i-1}\right|$, $\left|p', p'_{i+2}\right| = \left|p'_{i+1} p'_{i+2}\right|$, and the angle at p' is given by $\angle(p') = \angle(p'_{i-1}) + \angle(p'_{i+1})$. If the angle at p' is equal to the co-curvature at the corresponding point on Q, which can be checked with the corresponding local coordinate within a constant time, p' is placed into S as a new gluing point. Otherwise, p' is simply a new vertex on P'.

From the viewpoint of the spanning tree T, each gluing step from a gluing point in S corresponds to the removal of one leaf from T. Therefore, it is not difficult to see that the algorithm operates correctly.

Theorem 12 *Let $P' = (p'_0, p'_1, \ldots, p'_{n'-1}, p'_{n'} = p'_0)$ be the polygon given by the stamping. Then, the gluing check can be done in $O(n')$ time.*

Proof. Each gluing process decreases at least one edge from P' within a constant time. Therefore, the running time of the gluing check is $O(n')$ time. $\square$

3.5 Initial positions

In this section, we give the way to determine initial positions of P and Q.

By the discussion in Section 3.2, to determine initial positions of P and Q can be reduced into choices of $q \in V(Q)$, $f \in F(Q)$, $p, p' \in \partial(P)$ and $(p_x, p_y), (p'_x, p'_y) \in \mathbb{R}^2$. The detail on how to enumerate the pairs of (p, p') is given in Section 3.5.1. The detail for $(p_x, p_y), (p'_x, p'_y)$ is given in Section 3.5.2. The framework of ***EnumPosi*** is summarized in Algorithm 2.

Algorithm 2: ***EnumPosi***

Input : A polygon P and a polyhedron Q
Output: All possible initial positions of P and Q
foreach $\boldsymbol{q \in V(Q)}$ **do**
  foreach $\boldsymbol{f \in F(Q)}$ ***such that q is a vertex of f*** **do**
    Enumerate possible pairs (p, p') in $\partial(P)$.
    foreach $\boldsymbol{(p, p')}$ **do**
      Decide the coordinate of p, p' by ***DetermineDirection***.
      Output: the pair of the initial positions of P and Q

3.5.1 Choice of p, p'

In order to use later in Section 3.5.2, we require that p, p' satisfy $f_T(p), f_T(p') \in V(Q)$, and $f_T(p) \neq f_T(p)$.

Specifically, we use different ways depending on whether Q is a tetramonohedron. We first clarify the difference between tetramonohedron and the others as the following lemma.

Lemma 13 *Let Q be a convex polyhedron. When Q is not a tetramonohedron, Q has at least two vertices of curvature not equal to π.*

Proof. Let $q_0, \ldots, q_k$ be the vertices of Q. Because Q is a convex polyhedron, $k \geq 3$. When $k = 3$, the only possible solid is a doubly covered triangle. Thus Q satisfies the claim. If $k > 4$, by Theorem 1, at least two vertices have a curvature not equal to π. Thus, we focus on the case of $k = 4$. Through Lemma 5, because Q is not a tetramonohedron, four vertices cannot have curvatures equal to π. Based on the Gauss-Bonnet Theorem, it is impossible for three vertices to have a curvature of π except one.

Thus, Q has at least two vertices q and q' whose curvature is not equal to π. □

Combining Lemma 5 and Theorem 1, we have the following corollary.

Lemma 14 *Let Q be a convex polyhedron that is not a tetramonohedron, and P be an unfolding of Q. Then, P has at least two vertices of an angle not equal to π that correspond to distinct vertices of Q. Moreover, if Q has no vertex of curvature π, P has at least two vertices such that each vertex on P is glued to the corresponding vertex of Q without an extra angle from P.*

Proof. By Lemma 13, Q has at least two vertices q and q' whose curvature is not equal to π. Then, q and q' correspond to distinct vertices of P. Now, consider the set S_q of the vertices of P that are glued together to form q. Then, because the curvature of q is not equal to π and less than 2π, at least one of the elements in S_q has an angle unequal to π. Thus, q produces at least one vertex on $\partial(P)$ of an angle not equal to π. Based on the same argument, we have another vertex on $\partial(P)$ produced by q'.

Now, we assume that Q has no vertex of curvature π. Each leaf of the cutting tree corresponds to a vertex of Q, and this vertex forms a vertex of P because the curvature is not π. Because any tree has at least two leaves, we have the claim of this lemma. □

Therefore, when Q is not a tetramonohedron, it is sufficient that we consider each pair of the vertices of P as (p, p'). Thus, we can check all combinations in $O(n^2)$ time.

By contrast, when Q is a tetramonohedron, we may be unable to find any vertex of Q in the set of vertices of P. We can find a pair (p, p') in $\partial(P)$ by special tricks in Appendix A. For a given polygon P, we apply three Algorithms 5, 6, and 7 individually. These algorithms also take $O(n^2)$ time.

In both cases, we can choose (p, p') such that $f_T(p) = q, f_T(p') = q'$ for some $q, q' \in V(Q)$.

3.5.2 Determining coordinates of p and p'

By the discussion in Section 3.5.1, we can assume $(p_x, p_y) = (0, 0)$ because $f_T(p) = q$ and $(q_x, q_y, q_z) = (0, 0, 0)$.

We have the following theorem about (p'_x, p'_y).

Theorem 15 *Let $p_i, p_{i'}$ be two vertices of P, and $q_j, q_{j'}$ be two vertices of a commensurate convex polyhedron Q with respect to $\ell_{\min}$ and $\theta_{\min}$. It is then sufficient to check $O\left(\left(dL + n + \frac{2\pi}{\theta_{\min}}n\right)^{c-1}\right)$ combinations of (p'_x, p'_y) by Algorithm 3, where $d = \frac{c}{(1+c\sin\theta_{\min})\ell_{\min}}$.*

Moreover, we can modify the algorithm for the case that Q is a tetramonohedron, and the number of combinations is reduced into $O(L+n)$.

Proof. In the process of ***Stamp***, we focus on the part of steps starting at $p_i = q_j$ and ending at $p_{i'} = q_{j'}$ (Figure 3.5).

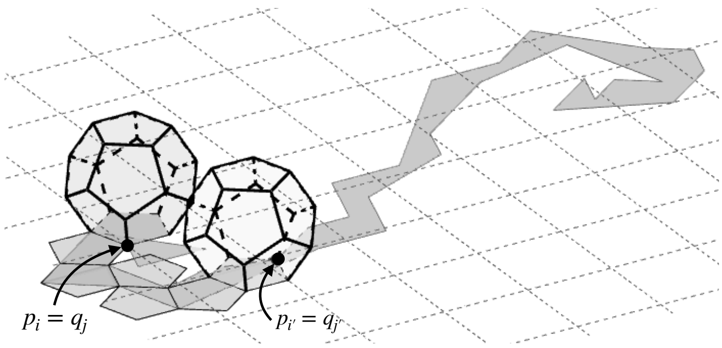

Figure 3.5: Rolling Q starting at point $p_i = q_j$ to the point $p_{i'} = q_{j'}$ on P in 2D and 3D

We then obtain the sequence of edges $e_0, e_1, \ldots, e_k$ of Q such that the corresponding vectors satisfy $\vec{e_0} + \vec{e_1} + \cdots + \vec{e_k} = p_i\vec{p}_{i'}$. It is sufficient to check all possible vectors.

Let $\vec{u_s} = (\ell_{\min}\cos s\theta_{\min}, \ell_{\min}\sin s\theta_{\min})$ for $s = 0, 1, \ldots, c-1$, where c is the minimum integer that $c\theta_{\min}$ is a multiple of π.

Then, based on the assumption of Q, we can observe that each vector $\vec{e_t}$ can be represented by a linear equation $\sum_{w=0}^{c-1} b_w \vec{u_w}$ for some integers $b_0, b_1, \ldots, b_{c-1}$.

Thus, through the commutativity of vectors, we have

$$p_i\vec{p}_{i'} = \vec{e_0} + \vec{e_1} + \cdots + \vec{e_k} = \sum_{w=0}^{c-1} B_w \vec{u_w}$$

for some integers $B_0, B_1, \ldots, B_{c-1}$. Now, we can see that $|B_w| < S$, where S is the number of rollings achieved during the stamping process. Based on the analysis, we can use the following algorithm to enumerate all possible coordinates (p'_x, p'_y).

Algorithm 3: DetermineDirection

Input : A polygon P, a polyhedron Q, two vertices p_i and $p_{i'}$ of P, and a vertex q_j of Q

Output: All possible combinations of (p'_x, p'_y)

for $B_0 \leftarrow -S$ **to** S **do**
  for $B_1 \leftarrow -S$ **to** S **do**
    ...
    for $B_{c-1} \leftarrow -S$ **to** S **do**
      if $|\vec{p_i p_{i'}}| = |\sum_{w=0}^{c-1} B_w \vec{u_w}|$ **then**
        $(p'_x, p'_y) = \sum_{w=0}^{c-1} B_w \vec{u_w}$

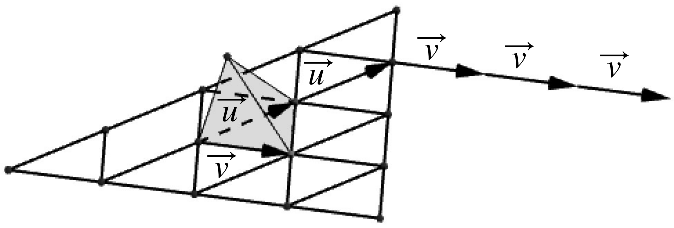

Figure 3.6: Two vectors u, v along the two edges of f

The number of iterations of the main loop of this algorithm is $O(S^c)$. However, the last parameter B_{c-1} can be determined within a constant time when the values of integers $B_0, B_1, \ldots, B_{c-2}$ are fixed. Thus, the last loop for B_{c-1} can be reduced. We, therefore, have the claim of the theorem. The number S is bounded by Theorem 11.

When Q is a tetramonohedra, the faces of Q make a triangle lattice. We can modify the algorithm by replacing $\vec{u_s}$ into the two vectors u, v along the two edges of f that incident to v (Figure 3.6).

□

3.6 Time complexity

We summarize the time complexity of the algorithms. The time complexity depends on whether Q is a commensurate convex polyhedron or a tetramonohedron.

In the process of ***EnumPosi***, we first choose a vertex q of Q, a face f, and a pair (p, p') of points of $\partial(P)$. It causes $O(m^2n^2)$ or $O(n^2)$ combinations.

After then, we determine the coordinate (p'_x, p'_y) of p'. By Theorem 15, it cause $O\left(\left(dL + n + \frac{2\pi}{\theta_{\min}} n\right)^{c-1}\right)$ or $O(L+n)$ combinations.

For each initial position, we do ***Stamp***. it takes $O\left(\left(dL + n + \frac{2\pi}{\theta_{\min}} n\right) m\right)$ or $O\left(L+n\right)$ time by Theorem 11.

Then, we do ***GlueCheck***. By Theorem 12, it takes $O(n')$ time. By the proof of Theorem 11, $n' < dL + n$ or $n' < L + n$.

Therefore, the total running time is

$$O\left(n^2 m^2 \times \left(dL + n + \frac{2\pi}{\theta_{\min}} n\right)^{c-1} \times \left(\left(dL + n + \frac{2\pi}{\theta_{\min}} n\right) m + (dL + n)\right)\right)$$

or

$$O\left(n^2 \times (L+n) \times ((L+n) + (L+n))\right)$$

which is simplified into $O\left(n^2 m^3 \left(dL + n + \frac{2\pi}{\theta_{\min}} n\right)^c\right)$ or $O(n^2(L+n)^2)$ time.

Chapter 4

Reconfiguration problem on refolding

In this chapter, we focus on a sequence of refolding steps. A polyhedron is ***refoldable*** into another polyhedron if we can unfold the original polyhedron into a polygon and fold the polygon into the target polyhedron. In other words, Q is refoldable into Q' if they share a common unfolding.

We show that several classes of polyhedra are connected by sequences of $O(1)$ refolding steps. In this chapter, we assume that any polyhedra have the same surface areas. More precisely, we prove that (1) any two tetramonohedra are refoldable into each other, (2) any doubly covered triangle is refoldable into a tetramonohedron, (3) any tetrahedron has a 3-step refolding sequence to a tetramonohedron, (4) any (augmented) regular prismatoid and doubly covered regular polygon are refoldable into a tetramonohedron, and (5) the regular dodecahedron has a 4-step refolding sequence to a tetramonohedron. In five platonic solids, the regular tetrahedron is included in tetramonohedra. Moreover, the regular hexahedron, regular octahedron, and regular icosahedron are included in augmented regular prismatoids. Thus, we obtain a $\leq$ 6-step refolding sequence between any pair of Platonic solids via tetramonohedra, applying (5) for the dodecahedron and (1) and/or (2) for all other Platonic solids.

4.1 Settings

We define a ***(k-step) refolding sequence*** from Q to Q' to be a sequence of convex polyhedra $Q = Q_0, Q_1, \ldots, Q_k = Q'$ where each Q_{i-1} is refoldable into Q_i. We refer to k as the ***length*** of the refolding sequence. To avoid confusion, we use "1-step refoldable" to refer to the previous notion of refoldability.

4.2 Connectivity of tetramonohedra

In this section, we show that any pair of tetramonohedra can be refolded to each other. Specifically, we show the following theorem.

Theorem 16 *For any $Q, Q' \in \Pi_4$, Q is 1-step refoldable into Q'.*

Proof. Let T be any triangular face of Q. Let a be the length of the longest edge of T and b be the height of T for the base edge of length a. We define T', a', and b' in the same manner as Q'; refer to Figure 4.1. We assume $a > a'$ without loss of generality. Now we have $a' > b'$ because a' is the longest edge of T', and $a'b' = ab$ because T and T' are of the same area. Thus, $(a')^2 = a'b'\frac{a'}{b'} > a'b' = ab$, and $2a' > a' > b$ by $a > a'$.

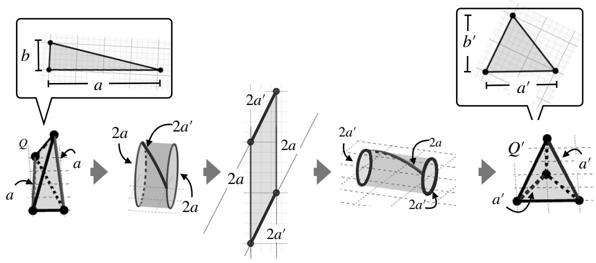

Figure 4.1: A refolding between two tetramonohedra

We cut two edges of Q of length a, resulting in a cylinder of height b and circumference $2a$. Then we can cut the cylinder by a segment of length $2a'$ because $2a' > b$. The resulting polygon is a parallelogram such that two opposite sides have length $2a$ and the other two opposite sides have length $2a'$(Figure 4.1). Now we glue the sides of length $2a$ and obtain a cylinder of height b' and circumference $2a'$. Then we can obtain Q' by folding this cylinder suitably (the opposite of cutting two edges of Q' of length $2a'$). $\square$

4.3 Polyhedral classes with a small number of vertices that have a finite step refolding sequence to a tetramonohedron

In this section, we show that any pair of polyhedra with ≤ 4 vertices have a finite step refolding sequence.

4.3.1 Doubly covered triangles

First, we focus on doubly covered triangles.

Theorem 17 *Any doubly covered triangle Q is 1-step refoldable into a doubly covered rectangle. Thus, Q has a refolding sequence to any doubly covered triangle Q' of length at most 3. If doubly covered triangles Q and Q' share at least one edge length, then the sequence has a length of at most 2.*

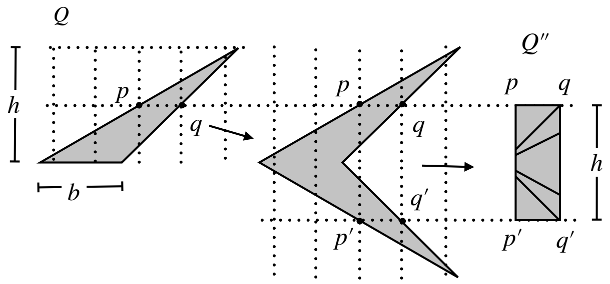

Figure 4.2: A refolding from a doubly covered triangle to a doubly covered rectangle

Proof. Let Q consist of a triangle T and its mirror image T'. We first cut Q along any two edges and unfold along the remaining attached edge, resulting in a quadrilateral unfolding as shown in Figure 4.2. Let b be the length of the uncut edge, which we call the ***base***, and let h be the height of T with respect to the base. Let p and q be the midpoints of the two cut edges. Then the line segment pq is parallel to the base and of length $b/2$. In the unfolding of Q, let p' and q' be the mirrors of p and q, respectively. Then we can draw a grid based on the rectangle $pp'q'q$ as shown in Figure 4.2. By folding along the crease lines defined by the grid, we can obtain a doubly covered rectangle Q'' of size $b/2 \times h$ (matching the doubled surface area of Q). (Intuitively, this folding wraps T and T' on the surface of the rectangle $pp'q'q$.)

Because Q'' is also a tetramonohedron, the second claim follows from Theorem 16. When Q has an edge of the same length as an edge of Q', as in the third claim, we can cut the other two edges of Q and Q' to obtain the same doubly covered rectangle, resulting in a 2-step refolding sequence. $\square$

The technique in the proof of Theorem 17 works for any doubly covered triangle Q, even if its faces are acute or obtuse triangles.

4.3.2 Tetrahedra

In this section, we prove that any tetrahedron can be refolded to a tetramonohedron.

Prior to the proof, we give the following two lemmas. Let $\mathcal{Q}_k$ denote the class of polyhedra with exactly k vertices.

Lemma 18 *Any polyhedron $Q' \in \Pi_2 \cap \mathcal{Q}_5$ is 1-step refoldable into a polyhedron $Q'' \in \Pi_3 \cap \mathcal{Q}_5$.*

Proof. Let λ_0, λ_1 be the two smooth vertices of Q'. We cut Q' by the shortest line segment l joining λ_0 and λ_1 and denote the obtained surface by C. By making crease lines from each of the other vertices to l perpendicularly and embedding the cut end of C to xy plane, we can form C as a triangular prism sliced (Figure 4.3). Let $h(t)$ be the height of a point t on the side of Q'. Let v_0, v_1, v_2 be the other vertices of Q' clockwise from the viewpoint of the outside of Q', and l_i be the shortest line segments from v_i to v_0. We assume that $h(v_0) \leq h(v_1), h(v_2)$ without loss of generality. Let θ_i denote the angle from the perpendicular line of v_0 to l_i. Then, since $\frac{\pi}{2} \leq \theta_1, \theta_2$ and $\theta_1 + \theta_2 < \sigma(v_0)$, we have $\kappa(v_0) < \pi$.

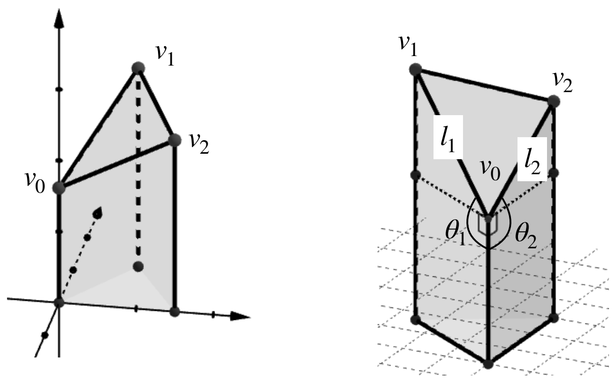

Figure 4.3: A triangular prism sliced diagonally

Since $\kappa(v_0)+\kappa(v_1)+\kappa(v_2) = 2\pi$ from Theorem 1, at least one of $\kappa(v_1), \kappa(v_2)$ is less than π. Thus we assume $\kappa(v_1) < \pi$. Let l' be the line where the counter-clockwise angle from l_1 to l' at v_0 is π (Figure 4.4). Note that the clockwise angle from l_1 to l' at v_0 is $\sigma(v_0) - \pi = \pi - \kappa(v_0)$.

By $h(v_0) \leq h(v_1), h(v_2)$, l' and l have an intersection point m. Let θ be the counter-clockwise angle from l' to l at m. l_1 and l' do not intersect except at v_0 because $^\forall t_1 \in l_1, ^\forall t_2 \in l'$ and $h(t_2) < h(v_0) < h(t_1)$. Then we cut C by l_1, l'(Figure 4.5).

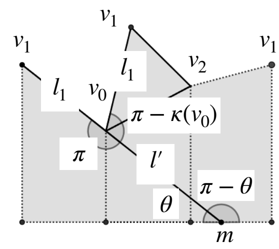

Figure 4.4: A side view of C

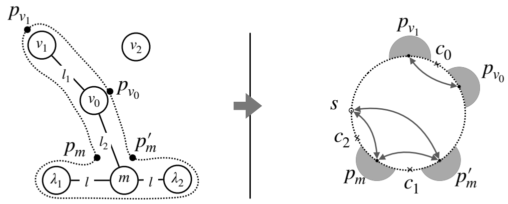

Figure 4.5: A way of cutting and gluing

On the obtained boundary, there are four points whose interior angles are not π: Let p_{v_0}, p_{v_1} correspond to v_0, v_1 and the both of p_m, p'_m correspond to m such that $\angle(p_{v_1}) = \sigma(v_1), \angle(p_{v_0}) = \pi - \kappa(v_0), \angle(p_m) = \theta$, and $\angle(p'_m) = \pi - \theta$. Let c_0 be the center point of (p_{v_1}, p_{v_0}), and s be the point that has the same distance with p'_m from c_0. Let c_1 and c_2 be the center point of $(p_m, p_{m'})$ and (p_m, s). We glue each of p_{v_0}, p_{v_1} and p'_m, p_m, s. Let Q'' be the resulting polyhedron. Since $\angle(p_{v_0}) + \angle(p_{v_1}) = \pi - \kappa(v_0) + \sigma(v_1) = 3\pi - (\kappa(v_0) + \kappa(v_1)) = 3\pi - (2\pi - \kappa(v_1)) = \pi + \kappa(v_1) < 2\pi$, Q'' satisfies Alexandrov's conditions. That is, Q'' is a convex polyhedron in $\Pi_3 \cap \mathcal{Q}_5$. □

Lemma 19 *Any polyhedron $Q'' \in \Pi_3 \cap \mathcal{Q}_5$ is 1-step refoldable into a polyhedron $Q''' \in \Pi_4$.*

Proof. Let $\lambda_0, \lambda_1, \lambda_2$ be the three smooth vertices of Q'' and v_0, v_1 be the other vertices. In the proof of Lemma 18, the vertex v_3 of Q' remains as the vertex of Q' without cutting, and v_0, v_1 are chosen such that $\kappa(v_0), \kappa(v_1) < \pi$ holds. Thus, we can apply the proof of Lemma 18 to a proof of Lemma 19 by replacing v_2 to λ_2. As a result, we obtain a polyhedron Q''' of Π_4. □

Under the above preparation, we show the subject theorem.

Theorem 20 *For any $Q \in \mathcal{Q}_4$, there is at most 3-step refolding sequence from Q to some $Q''' \in \Pi_4$.*

Proof. There are three possible cases of Q: $Q \in \Pi_0$, $Q \in \Pi_1$, or $Q \in \Pi_2$ (the case of $Q \in \Pi_3$ never happens by Theorem 1). First, we consider the case of $Q \in \Pi_0$ and $Q \in \Pi_1$. In each case, there are two vertices v, v' such that $\sigma(v), \sigma(v') \leq \pi$ by Theorem 1. We cut along the segment vv' and glue the point v to v'. On the resulting polyhedron, there is a new vertex of a co-curvature $\sigma(v) + \sigma(v') \leq 2\pi$ and two new smooth vertices. Thus, we can obtain a convex polyhedron $Q' \in \Pi_2$ by Theorem 4. That is, we can reduce these two cases to the case of $Q \in \Pi_2$ by the 1-step refolding. $Q' \in \Pi_2$ is 2-step refoldable into a polyhedron $Q''' \in \Pi_4$ by Lemma 18 and Lemma 19. □

4.4 Polyhedral classes that have a multi-step refolding sequence to a tetramonohedron

In this section, we show specific ways of unfolding and folding between some polyhedral classes and tetramonohedra.

4.4.1 Regular prismatoids

We focus on a polyhedral class called regular prismatoids. We extend the approach of Horiyama and Uehara [12], who showed that the regular icosahedron, the regular octahedron, and the regular hexahedron (cube) could be 1-step refolded into a tetramonohedron. As an example, Figure 4.6 shows their common unfolding for the regular icosahedron. A polygon $P = (p_0, c_1, p_1, c_2, p_2, \ldots, p_{2n}, c_{2n}, p_{2n+1}, p_0)$ is called a ***spine polygon*** if it satisfies the following two conditions (refer to Figure 4.7):

1. Vertex p_i is on the line segment p_0p_n for each $0 < i < n$; vertex p_i is on the line segment $p_{n+1}p_{2n+1}$ for each $n+1 < i < 2n+1$; and the polygon $B = (p_0, p_n, p_{n+1}, p_{2n+1}, p_0)$ is a parallelogram. We call B the ***base*** of P, and require it to have a positive area.

2. The polygon $T_i = (p_i, c_{i+1}, p_{i+1}, p_i)$ is an isosceles triangle for each $0 \leq i \leq n-1$ and $n+1 \leq i \leq 2n$. The triangles $T_0, T_1, \ldots, T_{n-1}$ are congruent, and $T_{n+1}, T_{n+2}, \ldots, T_{2n}$ are also congruent. These triangles are called ***spikes***.

Lemma 21 *Any spine polygon P can be folded to a tetramonohedron.*

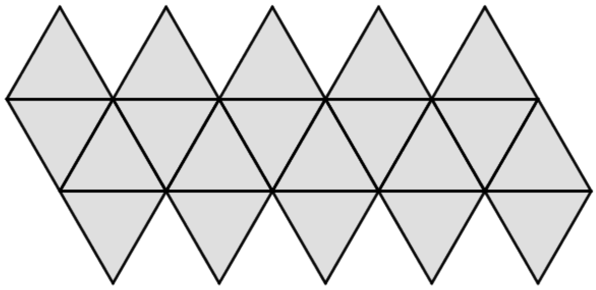

Figure 4.6: A common unfolding of a regular icosahedron and a tetramonohedron, from [12]

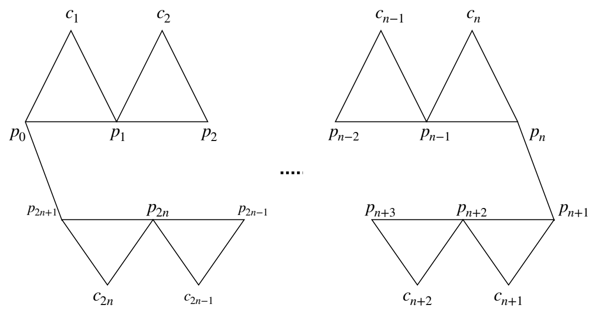

Figure 4.7: A spine polygon with $2n$ spikes

Proof. By Theorem 6, a polygon P can be folded into a tetramonohedron if the boundary of P can be divided into six parts, two of which are parallel and the other four of which are rotationally symmetric. We divide the boundary of a spine polygon P into $l_1 = (p_0, c_1, \ldots, c_n)$; $l_2 = (c_n, p_n)$, $l_3 = (p_n, p_{n+1})$; $l_4 = (p_{n+1}, c_{n+1}, \ldots, c_{2n})$, $l_5 = (c_{2n}, p_{2n+1})$; and $l_6 = (p_{2n+1}, p_0)$. Then l_3 and l_6 are parallel because the base of P is a parallelogram. Each of l_2 and l_5 is rotationally symmetric on its own as line segments, centering its midpoint. Each of l_1 and l_4 is rotationally symmetric because each spike of P is an isosceles triangle. □

Now we introduce some classes of polyhedra; refer to Figure 4.8.

A ***prismatoid*** is the convex hull of parallel ***base*** and ***top*** convex polygons. We sometimes call the base and the top ***roofs*** when they are not distinguished. We call a prismatoid ***regular*** if (1) its base P_1 and top P_2 are congruent regular polygons and (2) the line passing through the centers of P_1 and P_2 is perpendicular to P_1 and P_2. (Note that the side faces of a regular prismatoid do not need to be regular polygons.) The perpendicular distance between the planes containing P_1 and P_2 is the ***height*** of the prismatoid. The set of regular prismatoids contains ***prisms*** and ***antiprisms***, as well as doubly covered regular polygons (prisms of height zero). A ***pyramid*** is the convex hull of a ***base*** convex polygon and an ***apex*** point. We call a pyramid ***regular*** if the base polygon is a regular polygon, and the line passing through the apex and the center of the base is perpendicular to the base. (Note that the side faces of a regular pyramid do not need to be regular polygons.) A polyhedron is an ***augmented regular prismatoid*** if it can be obtained by attaching two regular pyramids to a regular prismatoid base-to-roof, where the bases of the pyramids are congruent to the roofs of the prismatoid, and each roof is covered by the base of one of the pyramids.

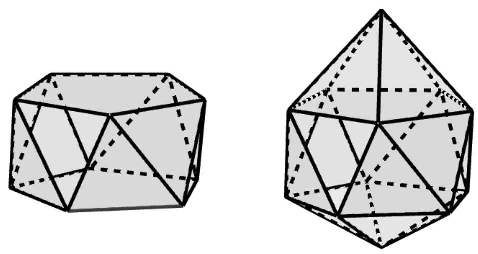

Figure 4.8: A regular prismatoid and an augmented regular prismatoid

Theorem 22 *Any regular prismatoid or augmented regular prismatoid of positive volume can be unfolded to a spine polygon.*

Proof. Let Q be a regular prismatoid. Let c_1 and c_2 be the center points of two roofs P_1 and P_2, respectively. Cutting from c_i to all vertices of P_i for each $i = 1, 2$ and cutting along a line joining between any pair of vertices of P_1 and P_2, we obtain a spine polygon. For an augmented regular prismatoid Q, we can similarly cut from the apex c_i of each pyramid to the other vertices of the pyramid, which are the vertices of the roof P_i of the prismatoid. □

When the height of the regular prismatoid is zero (or it is a doubly covered regular polygon), the proof of Theorem 22 does not work because the resulting polygon is not connected. In this case, we need to add some twists.

Theorem 23 *Any doubly covered regular n-gon is 1-step refoldable into a tetramonohedron for $n > 2$.*

Proof. First suppose that n is an even number $2k$ for some positive integer $k > 1$. We consider a special spine polygon where the top angles are $\frac{2\pi}{k}$; the vertices p_0, p_{2n+1}, p_1 are on a circle centered at c_1; and the vertices p_{2n+1}, p_1, p_2 are on a circle centered at c_{2n}; see Figure 4.9. Then we can obtain a doubly covered n-gon by folding along the zig-zag path $p_{2n+1}, p_1, p_{2n}, p_2, \ldots, p_{n+2}, p_n$ shown in Figure 4.9. Thus when $n = 2k$ for some positive integer k, we obtain the theorem.

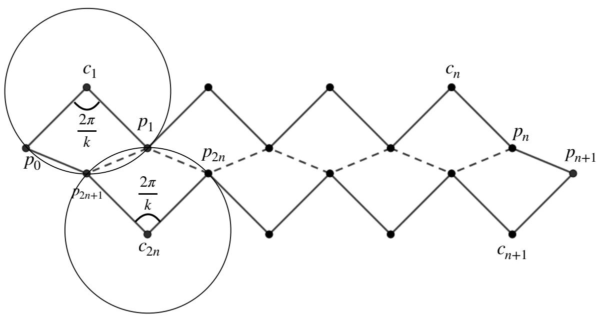

Figure 4.9: The case of a doubly covered regular 8-gon

Now suppose that n is an odd number $2k + 1$ for some positive integer k. We consider the spine polygon whose top angles are $\frac{4\pi}{2k+1}$; the vertices p_0, p_{2n+1}, p_1 are on a circle centered at c_1; and the vertices p_{2n+1}, p_1, p_2 are on a circle centered at c_{2n}. From this spine polygon, we cut off two triangles c_1, p_0, c_{2n+1} and c_{n+1}, p_{n+1}, p_n, as in Figure 4.10. Then we can obtain a doubly covered n-gon by folding along the zig-zag path $p_{2n+1}, p_1, p_{2n}, p_2, \ldots, p_{n+2}, p_n$ shown in Figure 4.10. Although the unfolding is no longer a spine polygon, it is easy to see that it can also fold into a tetramonohedron by letting

$l'_1 = (c_1, p_1, \ldots, p_n)$, $l'_2 = (p_n, p_n)$, $l'_3 = (p_n, c_{n+1})$, $l'_4 = (c_{n+1}, p_{n+2} \ldots, p_{2n+1})$, $l'_5 = (p_{2n+1}, p_{2n+1})$, and $l'_6 = (p_{2n+1}, c_1)$ in the proof of Lemma 21. $\square$

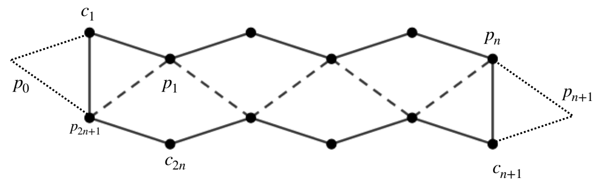

Figure 4.10: The case of a doubly covered regular 5-gon

The proof of Theorem 23 is effectively exploiting that a doubly covered regular $2k$-gon (with $k > 1$) can be viewed as a degenerate regular prismatoid with two k-gon roofs, where each of the side triangles of this prismatoid is on the plane of the roof sharing the base of the triangle.

Because the cube and the regular octahedron are regular prismatoids and the regular icosahedron is an augmented regular prismatoid, we obtain the following:

Corollary 24 *Let Q and Q' be regular polyhedra of the same area, neither of which is a regular dodecahedron. Then there exists a refolding sequence of length at most three from Q to Q'. When one of Q or Q' is a regular tetrahedron, the length of the sequence is at most 2.*

4.4.2 A regular dodecahedron

In this section, we show that there is a refolding sequence of the regular dodecahedron to a tetramonohedron of length 4.

Demaine et al. [8] mention that the regular dodecahedron can be refolded to another convex polyhedron. Indeed, they show that any convex polyhedron can be refolded to at least one other convex polyhedron using an idea called "flipping a Z-shape". We extend this idea.

Definition 25 *For a convex polyhedron Q and $n, k \in \mathbb{N}$, let $p = (s_1, s_2, \ldots, s_{(2k+1)n})$ be a path that consists of isometric and non-intersecting $(2k+1)n$ straight line segments s_i on Q. We cut the surface of Q along p. Then each line segment is divided into two line segments on the boundary of the cut. For each line segment s_i, let s_i^l and s_i^r correspond to the left and right sides on the boundary along the cut (Figure 4.11). Then p is a **Z-flippable (n, k)-path** on Q, and Q is **Z-flippable** by p, if the following gluing satisfies Alexandrov's conditions.*

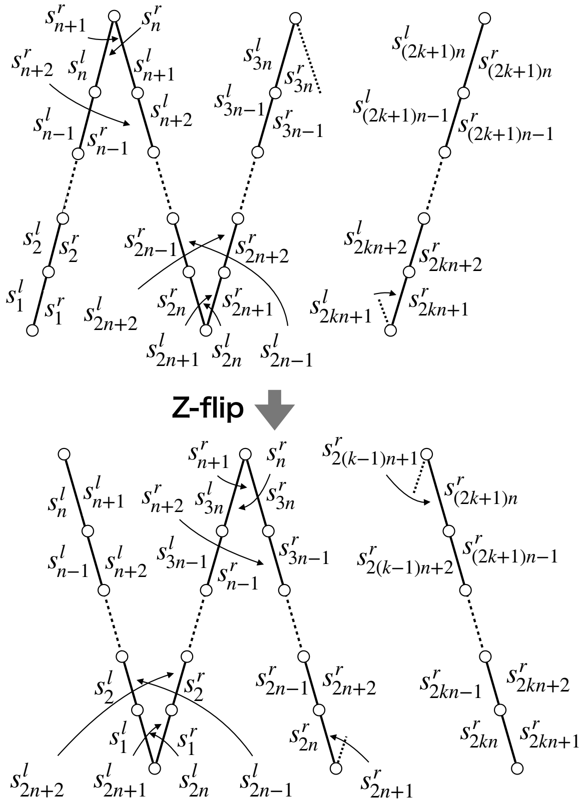

Figure 4.11: Z-flip

- *Glue* $s^l_1, s^l_2, \ldots, s^l_n$ *to* $s^l_{2n}, s^l_{2n-1}, \ldots, s^l_{n+1}$.
- *Glue* $s^r_1, s^r_2, \ldots, s^r_n$ *to* $s^l_{2n+1}, s^l_{2n+2}, \ldots, s^l_{3n}$.
- *Glue* $s^r_{n+1}, s^r_{n+2}, \ldots, s^r_{2n}$ *to* $s^r_{3n}, s^r_{3n-1}, \ldots, s^r_{2n+1}$.

$$\vdots$$

- *Glue* $s^r_{2(k-1)n+1}, s^r_{2(k-1)n+2}, \ldots, s^r_{2kn}$ *to* $s^r_{(2k+1)n}, s^r_{(2k+1)n-1}, \ldots, s^r_{2kn+1}$.

Figure 4.12 gives an example of a refolding by a Z-flippable $(1,1)$*-path.*

If there are Z-flippable paths $p^1, p^2, \ldots, p^m$ *inducing a tree structure on the surface of* Q*, we can flip them all at the same time (see Figure 4.13). Then we say that* Q *is* ***Z-flippable*** *by* $p^1, p^2, \ldots, p^m$*. This method also works when the obtained structure is disconnected trees with no intersections because we can flip each tree independently.*

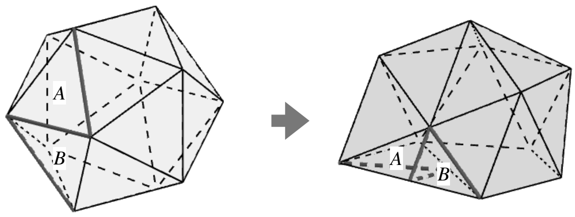

Figure 4.12: An example of Z-flippable $(1,1)$-paths

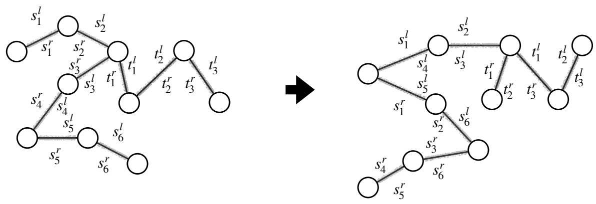

Figure 4.13: An example of Z-flippable paths that form a tree structure

Theorem 26 *There exists a 4-step refolding sequence between a regular dodecahedron and a tetramonohedron.*

Proof. Let D be a regular dodecahedron. To simplify, we assume that each edge of a regular pentagon is of length 1. We show that there exists a refolding sequence D, Q_1, Q_2, Q_3, Q_4 of length 4 for a tetramonohedron Q_4. All co-curvatures of the vertices of D are equal to $\frac{9\pi}{5}$. For any vertices v, there are 3 vertices of distance 1 from v and 6 vertices of distance $\phi = \frac{1+\sqrt{5}}{2}$ from v. Hereafter, in figures, each circle describes a non-flat vertex on a polyhedron, and the number in the circle describes its co-curvature divided by $\frac{\pi}{5}$. Each pair of vertices of distance 1 is connected by a solid line, and each pair of vertices of distance ϕ is connected by a dotted line. Figure 4.14 shows the initial state of D in this notation. We note that solid and dotted lines do not necessarily imply edges (or crease lines) on the polyhedron.

First, we choose $p^1 = (s_1^1, s_2^1, \ldots, s_6^1)$, $p^2 = (s_1^2, s_2^2, s_3^2)$, $p^3 = (s_1^3, s_2^3, \ldots, s_6^3)$, and $p^4 = (s_1^4, s_2^4, s_3^4)$ on the surface of D on the left of Figure 4.15. Then, p^1 and p^3 are Z-flippable $(2,1)$-paths, and p^2 and p^4 are Z-flippable $(1,1)$-paths. Thus, D is Z-flippable by p^1, p^2, p^3, p^4 to the polyhedron on the right of Figure 4.15. Let Q_1 be the resulting polyhedron.

Second, we choose $p^1 = (s_1^1, s_2^1, \ldots, s_5^1)$ on the surface of Q_1 on the left of Figure 4.16. Then, p^1 is a Z-flippable $(1,3)$-path. Thus, Q_1 is Z-flippable by p^1 to the next polyhedron Q_2 on the right of Figure 4.16.

Third, we choose $p^1 = (s_1^1, s_2^1, s_3^1)$ and $p^2 = (s_1^2, s_2^2, s_3^2)$ on the surface of Q_2 on the left of Figure 4.17. Then, p^1 and p^2 are Z-flippable $(1,1)$-paths.

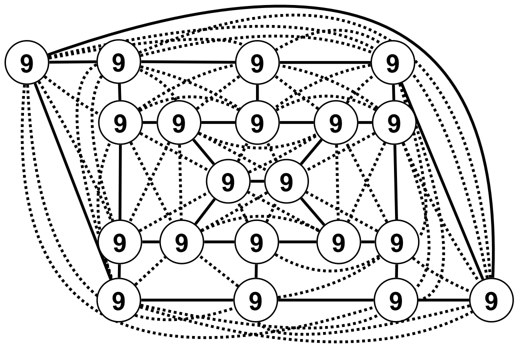

Figure 4.14: The initial regular dodecahedron

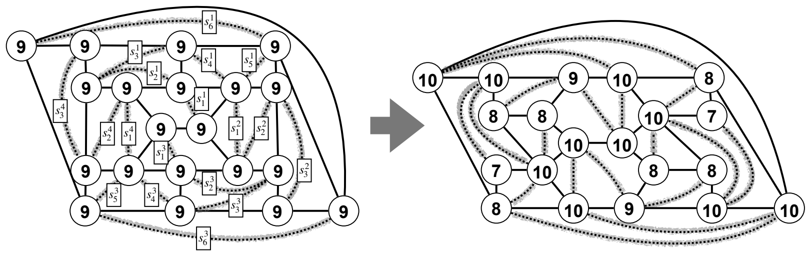

Figure 4.15: A refolding from D to Q_1

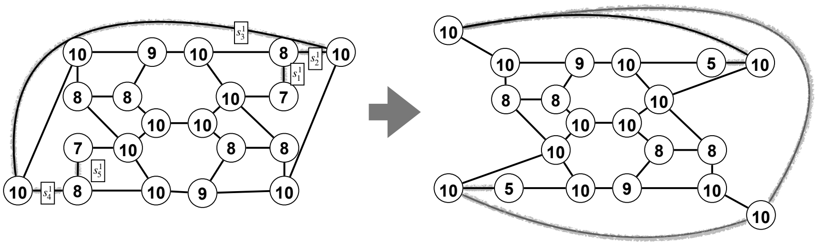

Figure 4.16: A refolding from Q_1 to Q_2

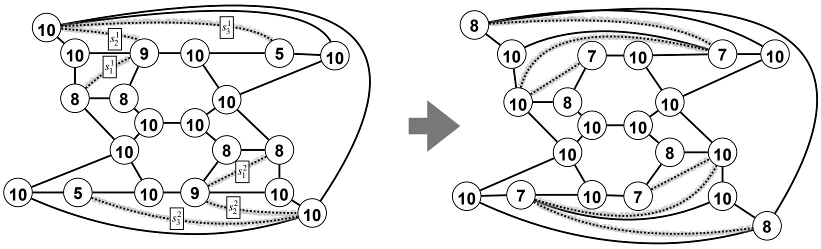

Figure 4.17: A refolding from Q_2 to Q_3

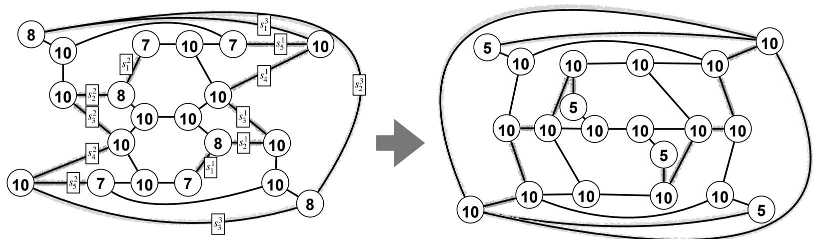

Figure 4.18: A refolding from Q_3 to Q_4

Thus, Q_2 is Z-flippable by p^1 and p^2 to the polyhedron Q_3 on the right of Figure 4.17.

Fourth, we choose $p^1 = (s^1_1, s^1_2, \ldots, s^1_5)$, $p^2 = (s^2_1, s^2_2, \ldots, s^2_5)$, and $p^3 = (s^3_1, s^3_2, s^3_3)$ on the surface of Q_3 on the left of Figure 4.18. Then, p^1 and p^2 are Z-flippable $(1,3)$-paths, and p^3 is a Z-flippable $(1,1)$-path. Thus, Q_3 is Z-flippable by p^1, p^2, and p^3 to the polyhedron Q_4 on the right of Figure 4.18.

Finally, we obtain a tetramonohedron Q_4 from a regular dodecahedron D by a 4-step refolding sequence. In this proof, we use partial unfolding between pairs of polyhedra in the refolding sequence.

We give the (fully unfolded) common unfoldings in Figures 4.19, 4.20, 4.21, and 4.22, which show the common unfoldings of each consecutive pair of polyhedra in the refolding sequence from the proof of Theorem 26. Thus, a 4-step refolding sequence exists between a regular dodecahedron and a tetramonohedron. □

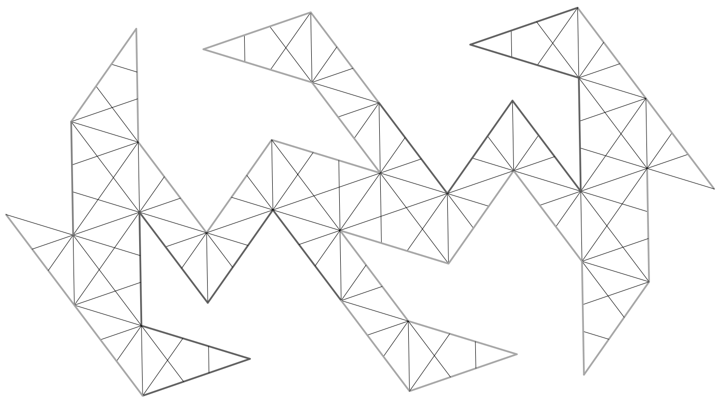

Figure 4.19: A common unfolding of D and Q_1

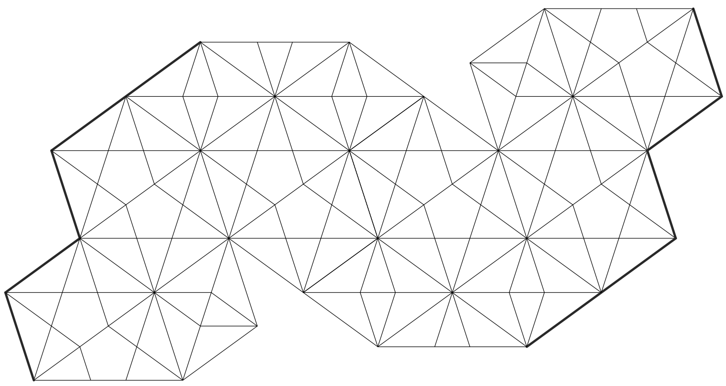

Figure 4.20: A common unfolding of Q_1 and Q_2

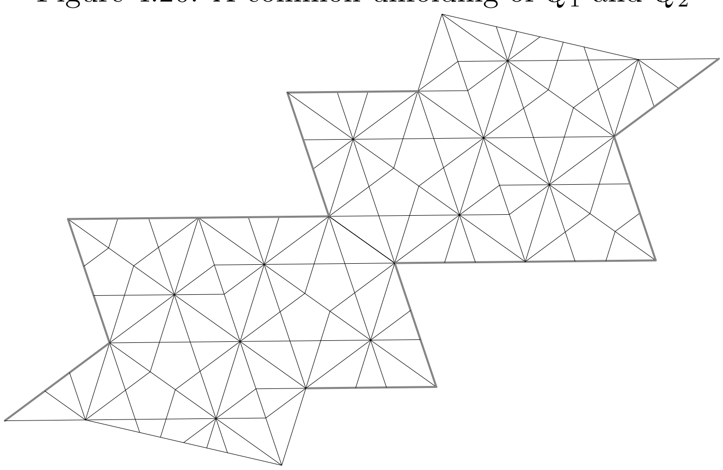

Figure 4.21: A common unfolding of Q_2 and Q_3

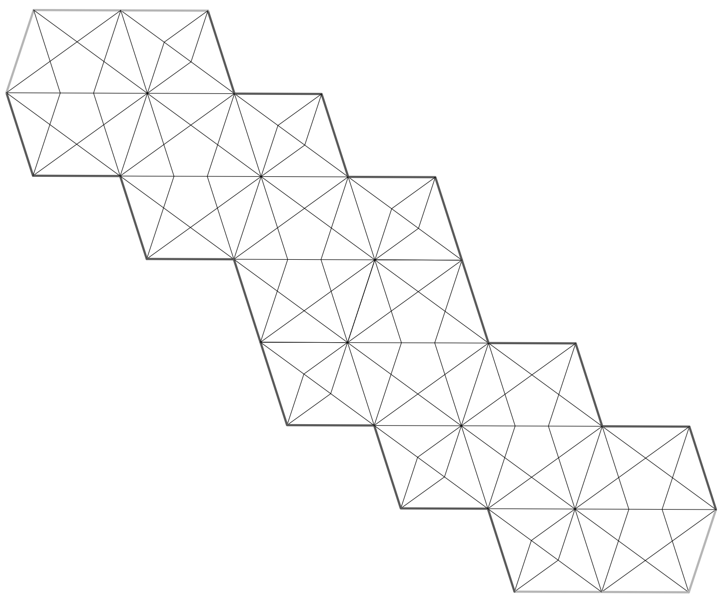

Figure 4.22: A common unfolding of Q_3 and Q_4

Chapter 5

Nonexistence of common unfolding

In this chapter, we consider the nonexistence of common unfoldings. We focus on relations of the interior angles of common unfoldings if they exist and introduce a notion called spreading trees. As a result, we give an algorithmic method for checking the existence of common unfoldings of a pair of polyhedra weakly-independent. Finally, we implement this algorithm and show that there is no common unfolding with k vertices within $k < 300$ between any strongly-independent and algebraic doubly covered triangles.

5.1 Settings

Let Q^0 and Q^1 be a pair of polyhedra with the same surface areas. We denote the set of the vertices of Q^i by $V(Q^i) = \{v_0^i, v_1^i, \ldots, v_{n^i-1}^i\}$ where n^i is the number of the vertices. We assume that Q^i can be unfolded to a polygon P^i by the cutting tree T^i and the folding map f^i. We define $D_k(T^i) := \{q \in T^i; deg(q) = k\}$.

Here, we obtain the following maps gl^i called ***gluing maps***, which represent the correspondence on the boundary $\partial(P^i)$.

Definition 27 *We define a gluing map $gl^i : \partial(P^i) \to \partial(P^i)$ by the map returns the point to which is glued by the mapping as follows (Figure 5.1).*

- *If $p \in D_2^i(P^i)$, $gl^i(p) := p'$ such that $f^i(p) = f^i(p')$; p' is determined uniquely.*
- *Otherwise, $gl^i(p) := p$.*

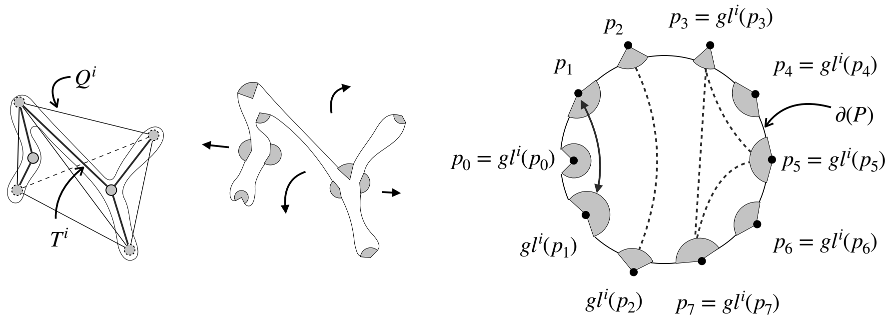

Figure 5.1: A schematic diagram of a gluing map

5.2 Spreading structures

If there is a common unfolding P of Q^0, Q^1, two gluing maps gl^0, gl^1 are defined on $\partial(P)$. For a point $p \in \partial(P)$, applying gl^{i+1} and gl^i alternately as $gl^i(p), gl^{i+1}(gl^i(p)), \ldots$, we obtain a sequence. In this section, we analyze the relationship that is induced by these sequences.

We define $L^i(P) := \{p \in \partial(P); f^i(p) \in D_1(T^i)\}$. Let take any point $p \in L^0(P)$, which is folded to a vertex v_j^0 when we fold P to Q^0. When we fold P to Q^1, we have three cases about $f^1(p)$. The first is the case of $f^1(p) \in D_1(T^1)$. In this case, p makes a vertex also when we fold P to Q^1. The second is the case of $f^1(p) \in D_2(T^1)$. In this case, p is glued to a point on $\partial(P)$ and makes a point on the surface of Q^1. The last is the case of $f^1(p) \in D_{>2}(T^1)$. In this case, p is glued to more than two points on $\partial(P)$. In the second case, by the definition of gl^i, we obtain a different point $gl^1(p)$ by applying gl^1 to p. Applying gl^0 to $gl^1(p)$, a similar division of case occurs. We can repeat the process until a point not in $D_2(P)$ is obtained. We define the points obtained in this process by a ***spreading path*** of p (Figure 5.2).

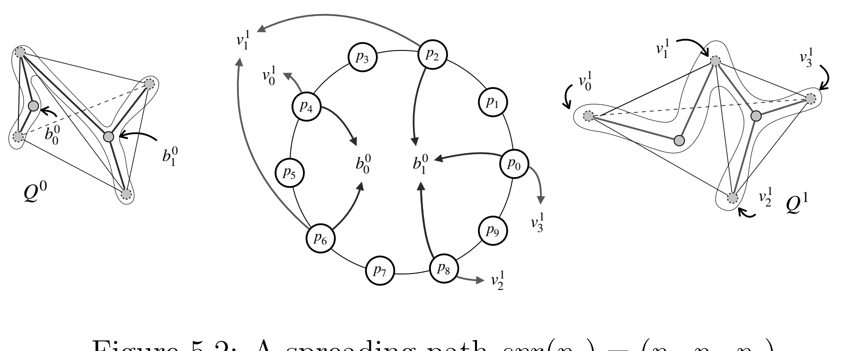

Figure 5.2: A spreading path $spr(p_4) = (p_4, p_6, p_2)$

More precisely, a spreading path is defined by the following.

Definition 28 *For a point $p \in L^i(P)$, we define the spreading path $spr(p)$ by the sequence of points of $\partial(P)$ obtained by alternative iterations of the map gl^{i+1} or gl^i,*

$$(p, gl^{i+1}(p), gl^i(gl^{i+1}(gl^i_j(p))), gl^{i+1}(gl^i(gl^{i+1}(p)), \dots),$$

until the map returns the same value as the input. In other words, a spreading path ends at a point in $D_{k\neq 2}(P)$. The endpoint is called a ***frontier*** *of $spr(p)$ and denoted by $frt_{spr(p)}$. Let $S^i(spr(p))$ be the set of the points in T^i that is spanned by $spr(p)$ except the frontier. Here we remark that $S^i(spr(p))$ would includes both a vertex and a point on the surface of Q^i; the co-curvature of a point in $S^i(spr(p))$ would be 2π or less.*

A spreading sequence has the following property about the interior angle of its frontier.

Lemma 29 *Let $spr(p)$ be a spreading path that has frontier $frt_{spr(p)}$ on T^i.*

$$\angle(frt_{spr(p)}) = (\sum_{q \in S^{i+1}(spr(p))} \sigma(q)) - (\sum_{q \in S^i(spr(p))} \sigma(q))$$

Proof. The point p, which is the first element of $spr(p)$, satisfies $\angle(p) = \sigma(v^i_j)$ where p is folded into a vertex v^i_j because of Observation 3. The point $gl^{i+1}(p)$, which is the second element of $spr(p)$, satisfies $\angle(gl^{i+1}(p)) = \sigma(f^{i+1}(p)) - \sigma(v^i_j)$ because of Observation 3. To repeat that, the angle at the frontier can be represented by the alternate sum of the co-curvatures at the points spanned by $spr(p)$. □

We compute the spreading sequences for all of $L^0(P) \cup L^1(P)$. By definition, each frontier is around a point in $D_{k\neq 2}(T^0) \cup D_{k\neq 2}(T^1)$. Inversely, each element of $D_{k\neq 2}(T^0) \cup D_{k\neq 2}(T^1)$ with degree k can be classified into three cases (1) with no frontiers, (2) at which $(< k-1)$ frontiers are gathering, (3) at which exactly $(k-1)$ frontiers are gathering, or (4) filled by k frontiers. In Case (3), we can consider a new spreading path beginning from the rest point which is not a frontier. We repeat this operation until no more points of Case (3). As a result, we obtain some connected components of spreading paths that are connected by added spreading paths ; each component forms a tree structure. We call them by ***spreading trees*** and denote by $st(\Lambda)$ where Λ is the set of the sub spreading paths that begin from an element of $L^0(P) \cup L^1(P)$. For a spreading tree $st(\Lambda)$, we call the points of Case (4) by ***internal vertices***, and the points of Case (2) by ***frontiers***.

A spreading tree is ***complete*** if it has no frontier. Let $S^i(st(\Lambda))$ be the union of $\bigcup_{\lambda\in\Lambda} S^i(\lambda)$ and all internal vertices of $st(\Lambda)$. $S^i(st(\Lambda))$ would include both vertices and points on surface of Q^i. We define $E^i(st(\Lambda)) := \{v \in S^i(st(\Lambda)); v \in V(Q^i)\}$ and $M^i(spr(p)) := \{v \in S^i(st(\Lambda); v \notin V(Q^i)\}$.

By Lemma 29, the following holds.

Observation 30 *For any complete spreading tree $st(\Lambda)$,*

$$(\sum_{q\in S^{i+1}(st(\Lambda))} \sigma(q)) - (\sum_{q\in S^i(st(\Lambda))} \sigma(q)) = 0.$$

To clarify the structure of the spreading trees, we introduce the following restriction on the co-curvatures of the vertices. Here we note that, for the co-curvatures, $(\sum_{v\in V(Q^i)} \sigma(v)) = 2\pi(n^i - 2)$ holds by Theorem 1.

Definition 31 *A pair of polyhedra Q^0 and Q^1 is **weakly-independent** if $^\forall m^i_j \in \{0,1\}, (\sum_{v_j\in V(Q^0)} m^0_j\sigma(v_j)) - (\sum_{v_j\in V(Q^1)} m^1_j\sigma(v_j))$ is not a multiple of π unless $m^i_0 = m^i_1 = \ldots = m^i_{n^i-1}$ for each i.*

We can determine the structure of spreading trees if a pair of polyhedra is weakly-independent by the following lemmas.

Lemma 32 *If there is a common unfolding of Q^0 and Q^1 which are weakly-independent, just two complete spreading trees $st(\Lambda^0)$ and $st(\Lambda^1)$ are defined such that $E^i(st(\Lambda^i))) = V(Q^i)$.*

Proof. Let $st(\Lambda^0), st(\Lambda^1), \ldots, st(\Lambda^s)$ be the spreading trees. By the definition of weakly-independence and Observation 30, a spreading tree $st(\Lambda)$ is complete if and only if $[E^i(st(\Lambda)) = V(Q^i)] \vee [E^i(st(\Lambda)) = \emptyset]$ for each i. Therefore, there are three possible cases; (1) $V(Q^0)$ and $V(Q^1)$ are spanned by one complete spreading tree, (2) $V(Q^0)$ and $V(Q^1)$ are spanned by different complete spreading trees, and (3) there are incomplete spreading trees. We show that only case (2) is possible.

Prior to consider the proof, we introduce a notation for tree graphs. For a tree graph G, let $l(G)$ be the number of leaves and $D(G)$ be the set of the vertices with ≤ 3 degree. Here, it is easy to see that the followings hold where G_0, G_1, G_2 are tree graphs.

- If there exists a one to one map $d : D(G_0) \to D(G_1)$ such that $deg(v) = deg(d(v))$, $l(G_0) = l(G_0)$ holds.
- $D(G_2) = D(G_0) \cup D(G_0) \implies l(G_2) = l(G_0) + l(G_1) - 1$

- $D(G_0) \subset D(G_1) \implies l(G_0) \leq l(G_1)$

We consider Case (1). Let $st(\lambda)$ be the complete spreading tree such that satisfies $[E^i(st(\Lambda)) = V(Q^i)]$ for each $i = 0, 1$. By the definition of $L^0(P) \cup L^1(P)$, $l(T^0) + l(T^1) = l(st(\Lambda))$. An internal vertex of $st(\lambda)$ with degree k is realized on a degree k point of the cutting trees T^0 or T^1. It means that there exists a map $d : D(st(\lambda)) \to D(T^0) \cup D(T^1)$ such that $deg(v) = deg(d(v))$. Thus, $l(st(\lambda)) < l(T^0) + l(T^1)$. It contradicts too $l(st(\lambda)) = l(T^0) + l(T^1)$. Therefore, we can conclude Case (1) is infeasible.

It is easy to see Case (3) is also infeasible because an incomplete spreading tree consumes more degrees at internal vertices than a complete one. □

As a corollary, the following also holds.

Lemma 33 *If Q^0 and Q^1 are weakly-independent,* $\forall p \in L^0(P) \cup L^1(P), frt_{spr(p)} \notin L^0(P) \cup L^1(P)$

Proof. We assume that there exist two points $p, p' \in L^0(P) \cup L^1(P), frt_{spr(p)} = p'$. Two spreading paths $spr(p)$ and $spr(p')$ are identical as a spreading tree $st(\lambda)$. By Lemma 32, p and p' are included on the same side of $L^0(P)$ or $L^1(P)$. We assume that $p, p' \in L^0(P)$ and p, p' are folded into vertices $v^0_j, v^0_{j'}$ without loss of generality. Here, $st(\lambda)$ satisfies $[V^0(st(\lambda)) = E^0(st(\lambda))] \wedge [V^1(st(\lambda)) \cap E^1(st(\lambda)) = \emptyset]$. When we consider the value of $(\sum_{q \in S^1(st(\lambda))} \sigma(q)) - (\sum_{q \in S^0(st(\lambda))} \sigma(q))$, it is reduced to $(2\pi \cdot |M^1(st(\lambda))|) - (2\pi(n^0 - 2) + 2\pi \cdot |M^0(st(\lambda))|)$ because $\sigma(q) = 2\pi$ for any $q \in M^i(st(\lambda))$ and Theorem 1.

Because $st(\lambda)$ is a path whose both end points are in $L^0(P)$, $|M^1(st(\lambda))| > |M^0(st(\lambda))|$ and the difference of them is $n^0 - 1$. As a result, $(\sum_{q \in S^1(st(\lambda))} \sigma(q)) - (\sum_{q \in S^0(st(\lambda))} \sigma(q)) = 2\pi$. It contradicts to Observation 30. □

Here, we introduce a class of common unfolding called a ***standard-form*** using spreading trees.

Definition 34 *A common unfolding between Q^0, Q^1 is a standard-form if all vertices of P are included in $\bigcup_i st(\Lambda^i)$.*

The following lemma shows that any standard-form common unfolding is an equilateral polygon.

Lemma 35 *If there is a standard-form common unfolding of Q^0 and Q^1 which are weakly-independent, the points in $\bigcup_i st(\Lambda^i)$ appear in $\partial(P)$ at even intervals. Moreover, the elements of $st(\Lambda^0)$ and $st(\Lambda^1)$ appear alternately.*

Proof. For a point $u \in \bigcup_i st(\Lambda^i)$, we define $d_+(u)$ and $d_-(u)$ by the counterclockwise and clockwise distances between u and the nearest point in $\bigcup_i st(\Lambda^i)$. We show that $d_+(u) = d_-(u) = c$ for any u where c is a constant value. Let p be a leaf point of $st(\Lambda^i)$ for a fixed $i = 0, 1$. The point p is cut by a leaf l of T^i. Obviously, $d_+(p) = d_-(p)$ holds because the nearest point l' of l on T^i makes both side nearest points p_+, p_- of p on $\partial(P)$ (Figure 5.3). Along $spr(p)$, the interval (p_+, p, p_-) is glued to another interval on $\partial(P)$ by gl^{i+1}, and the distances are kept (Figure 5.4). Therefore, we can confirm by induction that $d_+(u) = d_-(u) = c$ holds.

The rest is to show that the points of $st(\Lambda^i)$ are not adjacent for each i. We can see this by comparing the distances between the points in $st(\Lambda^0)$ and $st(\Lambda^0)$.

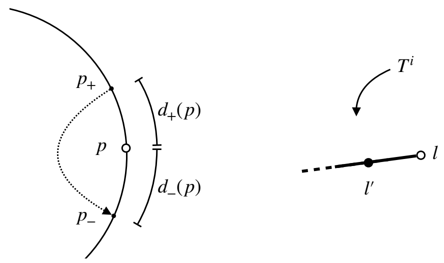

Figure 5.3: The both sides nearest points of p

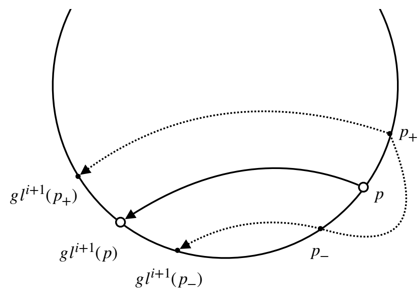

Figure 5.4: Spreading nearest distances

□

We can reduce the existence of common unfoldings to the existence of standard-form common unfoldings by the following lemma.

Lemma 36 *If there is a common unfolding of Q^0 and Q^1 that are weakly-independent, Q^0 and Q^1 have a standard-form common unfolding.*

Proof. By Lemma 35, the points in the spreading trees are lined up alternately on ∂P. Let take a pair of adjacent points and m be the interval between them. Let $(p_0, p_1, p_2, \ldots, p_k)$ be the vertices of P on m'. Because m is glued to another interval m', $(p_0, p_1, p_2, \ldots, p_k)$ make vertices $(p'_0, p'_1, p'_2, \ldots, p'_k)$ such that $\angle(p_i) = 2\pi - \angle(p'_i)$. In the same way as the proof of Lemma 35, it spreads into all intervals. On the boundary of P, except for the points in the spreading trees, the interior angles are $\angle(p_0), \ldots, \angle(p_k)$ and $2\pi - \angle(p_k), \ldots, 2\pi - \angle(p_0)$ alternately; see Figure 5.5. We focus on the cutting tree T into one side polyhedron. Let T' be the cutting tree replacing each interval of T with a straight line segment. T' has kept the interior angles at the points in the spreading trees; see Figure 5.6. Let P' be the unfolding by T'. Then P' is a standard-form common unfolding of Q^0 and Q^1. □

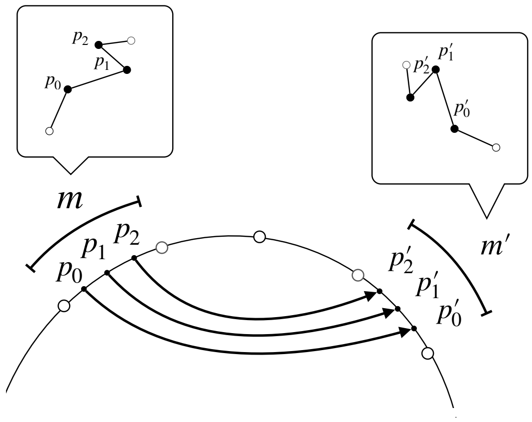

Figure 5.5: $(p_0, p_1, \ldots, p_m)$ on the interval m

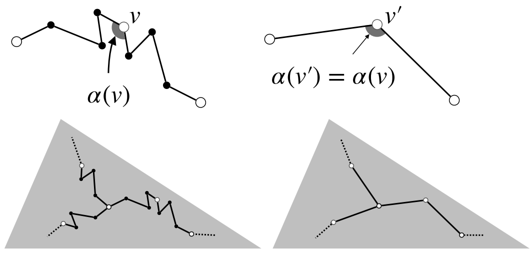

Figure 5.6: The reduction of a common unfolding into a standard-form common unfolding

From Lemma 35 and Lemma 36, we can conclude as the following theorem.

Theorem 37 *If there is a common unfolding of Q^0 and Q^1 that are weakly-independent, Q^0 and Q^1 have an equilateral common unfolding.*

5.3 Discretization of the existence of common unfoldings

In this section, we consider the existence of standard-form common unfoldings.

A standard-form common unfolding is an equilateral polygon whose interior angles are included in spreading trees. It means that a standard-form common unfolding can be represented by a sequence $\phi = (\phi_0, \ldots, \phi_n)$ of interior angle where each ϕ_i is a summation of co-curvatures $\sigma(v^i_j)$ and a multiple of 2π. Inversely, we consider restoring an equilateral polygonal line from a sequence of angles. For a sequence of interior angle $\phi = (\phi_0, \ldots, \phi_n)$, we define the polygonal line $Poly_\phi = (p_0, p_1, \ldots, p_n)$ in the complex plane $\mathbb{C}$ by the following.

$$p_0 = 1, p_1 = 0 \in \mathbb{C},$$
$$p_{i+1} - p_i = (p_{i-1} - p_i)e^{\sqrt{-1}\,\phi_i}.$$

Here, we remark that $e^{\sqrt{-1}\,\theta} = \cos\theta + \sqrt{-1}\,\sin\theta$ holds by Euler's Formula.

If there is a standard-form common unfolding, there must exist a sequence ϕ such that $Poly_\phi$ satisfies $p_0 = p_n$.

We consider a condition that $Poly_\phi$ satisfies $p_0 = p_n$. Here, we introduce more restricted polyhedral classes.

- A pair of polyhedra Q^0 and Q^1 are ***strongly-independent*** if ${}^\forall m^i_j \in \mathbb{Q}, (\sum_{v_j \in V(Q^0)} m^0_j \sigma(v_j)) - (\sum_{v_j \in V(Q^1)} m^1_j \sigma(v_j))$ is not a multiple of π unless $m^i_0 = m^i_1 = \ldots = m^i_{n^i-1}$ for each i.
- A polyhedron Q^i is ***algebraic*** if $\sigma(v^i_j) \in \mathbb{Q}^*$ for any $v^i_j \in V(Q^i) - \{v^i_0\}$ where $\mathbb{Q}^*$ is the algebraic closure on $\mathbb{Q}$.

We define that a polygonal line $Poly_\phi$ is parallel if there exists a permutation τ on $(\vec{w_0}, \vec{w_1}, \ldots, \vec{w_{n-1}})$ such that $(\tau(\vec{w_i}) = \vec{w_j} \implies \vec{w_i} = -\vec{w_j}) \wedge (\tau^2(\vec{w_i}) = \vec{w_i})$ where $\vec{w_i}$ is the vector along the edge (p_i, p_{i+1}).

Here, we state the condition that $Poly_\phi$ satisfies $p_0 = p_n$.

Lemma 38 *If ϕ is generated by a standard-form common unfolding of strongly-independent and algebraic polyhedra Q^0 and Q^1, $Poly_\phi$ satisfies $p_0 = p_n$ if and only if $Poly_\phi$ is parallel.*

Proof. If $Poly_\phi$ is parallel, $p_0 = p_n$ holds because each pair of $\vec{w}_i$ and $\tau(\vec{w}_i)$ is canceled when we trace the $Poly_\phi$. We show that its inverse holds. Here, $p_0 = p_n$ is equivalent to $\sum_i \vec{w}_i = 0$. We should show that the vectors $\vec{w}_i$ are linearly independent on $\mathbb{Z}$ except that two vectors are the same or inverse. Let $\vec{w}'_0, \vec{w}'_1, \ldots, \vec{w}'_k$ be the subset of $\vec{w}_0, \vec{w}_1, \ldots, \vec{w}_{n-1}$ choosing without the same or inverse ones. We use a classical result on transcendental numbers:

Theorem 39 (Lindemann's Theorem) *For any distinct algebraic numbers $a_0, a_1, \ldots, a_m$, the numbers $e^{a_0}, e^{a_1}, \ldots, e^{a_m}$ are linearly independent on $\mathbb{Q}^*$, where $\mathbb{Q}^*$ is the algebraic closure on $\mathbb{Q}$.*

Let ψ_i be the slope of $\vec{w}'_i$; $\vec{w}'_i$ is represented by $e^{\sqrt{-1}\,\psi_i}$. The slope ψ_i is $\phi_0 + \phi_1 + \cdots + \phi_i$ or $\phi_0 + \phi_1 + \cdots + \phi_i - \pi$ depending on whether i is odd or even. Because we choose $\vec{w}'_0, \vec{w}'_1, \ldots, \vec{w}'_k$ without the same or inverse ones, $\psi_0, \ldots, \psi_k$ are distinct algebraic numbers. Similarly, $\sqrt{-1}\,\psi_0, \ldots, \sqrt{-1}\,\psi_k$ are distinct algebraic numbers. By Lindemann's Theorem, $e^{\sqrt{-1}\,\psi_0}, \ldots, e^{\sqrt{-1}\,\psi_k}$ are linearly independent on $\mathbb{Q}^*$. On $\mathbb{Z}$, they are also linearly independent. Therefore, $\vec{w}'_0, \vec{w}'_1, \ldots, \vec{w}'_k$ are linearly independent on $\mathbb{Z}$. □

For a sequence ϕ generated by a pair of strongly-independent and algebraic polyhedra Q^0 and Q^1, the shape of $Poly_\phi$ depends on the concrete values of $\sigma(v^i_j)$. In contrast, the following holds because two vectors along edges of $Poly_\phi$ satisfy $\vec{w}_i = -\vec{w}_j$ only when $\phi_0 + \phi_1 + \cdots + \phi_i = 0$, and it is not depend on the concrete values of $\sigma(v^i_j)$ by the strongly-independence.

Remark 40 *If ϕ is generated by a standard-form common unfolding of strongly-independent and algebraic polyhedra Q^0 and Q^1, whether $Poly_\phi$ is parallel does not depend on the concrete values of $\sigma(v^i_j)$.*

We conclude what we can say if there is a k-gon P that is a standard-form common unfolding of a pair of strongly-independent and algebraic polyhedra. The boundary of P is glued in two ways by two gluing maps gl^0, gl^1. These gluing maps define on $\partial(P)$ two spreading trees are defined. Moreover, $Poly_\phi$ would be parallel where ϕ is the sequence of the interior angles of P.

Inversely, we can check the nonexistence of common unfoldings with k vertices for fixed k by enumerating all possible ways of gluing. We give a specific algorithm for a certain polyhedral class.

5.4 The case for doubly covered triangles

We show that there is no common unfolding with k vertices within $k < 300$ between two doubly covered triangles that are strongly-independent and algebraic.

The topology of a cutting tree T of a doubly covered triangle can be classified into two cases, as illustrated in Figure 5.7: a ***Y-form*** is a tree with a single point b^i of degree 3 (and with leaves at the vertices of Q), and a ***V-form*** is just a path (through all vertices of Q).

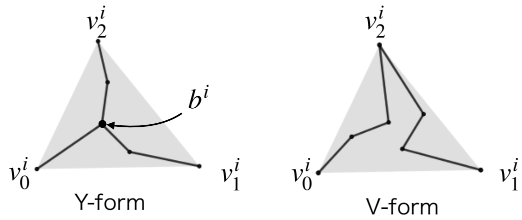

Figure 5.7: Topologies of cutting trees of doubly covered triangles

By Corollary 33, only the case that both the cutting trees are Y-form is possible. Therefore, there are three points l_0^i, l_1^i, l_2^i such that $f^i(l_j^i) = v_j^i$ and three points m_0^i, m_1^i, m_2^i such that $b^i = f^i(m_0^i) = f^i(m_1^i) = f^i(m_2^i)$.

By Remark 40, we do not care about the concrete values of $\sigma(v_j^i)$ and treat each $\sigma(v_j^i)$ as a symbol θ_j^i.

First, we prepare a cyclic array C of length k. Next, we enumerate all possible forms of T^i by determining the values of m_j^i. Then, we check whether the two spreading trees are defined well. Finally, we check whether the obtained sequence of the interior angles makes a parallel polygon. Details of the algorithm are given in Algorithm 4. It requires $O(k^7)$ time theoretically. We implemented them and checked that there is no common unfolding with k vertices in a range $k < 300$. It takes 1.5 hours in a normal laptop environment (CPU: 1.4GHz Intel Quad-Core i5, OS: mac OS 12.4, Memory: 16GB, compiler: GCC $11.3.0_2$, optimize: -O3).

Algorithm 4: Checking algorithm for doubly covered triangles

input : The number of vertices k
output: Whether there is a common unfolding with k vertices between two doubly covered triangles that are strongly-independent and algebraic.

```
Let C be a cyclic array of length k and m_0^0 := 0.
forall m_1^0, m_2^0, m_0^1, m_1^1, m_2^1 such that
 0 = m_0^0 < m_1^0 < m_2^0 < n, 0 < m_0^1 < m_1^1 < m_2^1 < n do
    for i = 0,1 and j = 0,1,2 do
        if m_{j+1}^i - m_j^i are odd then
            Return to line 2.
        l_j^i := m_j^i + (1/2)(m_{j+1}^i - m_j^i) mod n
    Define gl^0, gl^1.
    for i = 0,1 and j = 0, 1, 2 do
        p := l_j^i
        k := (j + 1) mod 2
        C[l_j^i] := θ_j^i.
        while p ≠ gl^k(p) do
            p := gl^k(p)
            if C[p] is not yet defined then
                if k = 1 then
                    C[p] := θ_j^i
                else
                    C[p] := \overline{θ_j^i}
            else
                Return to line 2.
            k := (k + 1) mod 2
    if {C[m_0^i], C[m_1^i], C[m_2^i]} = {θ_0^{i+1}, θ_1^{i+1}, θ_2^{i+1}} for each i then
        Let S be the stack of the numbers 0 to k.
        while S ≠ ∅ do
            for i ∈ S do
                if there is (i <)j ∈ S such that
                 C[i] + C[i+1] + ··· + C[j] is a multiple of 2π then
                    Remove i, j from S.
                else
                    Return to line 3.
        output: true
output: false
```

Chapter 6

Conclusion

In Chapter 3, we provided a pseudo-polynomial time algorithm for solving the folding problem for the given simple polygon P and convex polyhedron Q. To bind the running time of our algorithm, we introduced a new notion of a commensurate convex polyhedron with a unit length and a unit angle such that each edge has a length multiple of the unit length, and each angle of a face is a multiple of the smallest angle. The set of commensurate convex polyhedra includes wide classes of polyhedra; for example, Platonic solids, convex polyhedra formed by regular polygons (i.e., the 13 Archimedean and 92 Johnson solids), rational boxes, the 13 Catalan solids (also known as Archimedean dual), and so on. In fact, we can use our algorithm for a general convex polyhedron. However, in this case, we cannot bind the running time so we may need a different approach.

In Chapter 4, we gave a partial answer to Open Problem 25.6 in [9]. For every pair of regular polyhedra, we obtain a refolding sequence of length at most 6. Although this is the first refolding result for the regular dodecahedron, the number of refolding steps seems a bit large. Finding a shorter refolding sequence is an open problem. The notion of refolding sequence raises many open problems. Which pairs of convex polyhedra are connected by a refolding sequence of finite length? Is there any pair of convex polyhedra that are not connected by any refolding sequence? At the center of our results is that the set of tetramonohedra induces a clique by the binary relation of refoldability. Is the regular dodecahedron refoldable into a tetramonohedron? Are all Archimedean and Johnson solids refoldable into tetramonohedra? Is there any convex polyhedron that is not refoldable into a tetramonohedron?

In Chapter 5, we proved the nonexistence of common unfoldings limited in the number of vertices between two elements in a restricted polyhedral class. The main next step is to remove the limitation on the number of vertices of an unfolding. As we can see from the computational experiments, Lemma 38 requires a strong condition to have a common unfolding. This condition does not seem to be satisfied by any sequence obtained by Algorithm 4. If we can prove this conjecture, then we will obtain nonexistence without limiting the number of vertices. The notion of spreading structures gives us a reason why it is difficult to make a common unfolding for a given pair of polyhedra. Another next step is the extension of a spreading structure into non-independent pairs of polyhedra. Applying to Platonic solids would help to give the negative answer to Open Problem 25.6 in [9].

Appendix A

Special tricks for the case of tetramonohedra

Here, we show special tricks to find a pair (p, p') of points on the boundary of an unfolding of a tetramonohedron Q that is corresponded to a vertex of Q.

When a tetramonohedron Q is unfolded by a cutting tree T into a polygon P, leaves of T make π on $\partial(P)$ because each vertex of a tetramonohedron Q has a curvature π (Lemma 5). Thus, we cannot directly find the vertex of Q on a straight line of $\partial(P)$.

Let $V(Q) = \{v_1, v_2, v_3, v_4\}$. Because T is a tree, T contains at least two leaves in v_1, v_2, v_3, and v_4.

According to the case analysis in [2], T has one of the following topological structures, i.e., an X-shape, a Y-shape, a U-shape, an F-shape, or an H-shape (Figure A.1).

For these trees, we introduce certain notations. For each leaf ℓ of a cutting tree T, the ***associated edge*** $e(\ell)$ of ℓ is the unique edge incident to ℓ. When $e(\ell) = \{u, \ell\}$, u is the ***parent*** of ℓ. On T, $a_R(\ell)$ and $a_L(\ell)$ are the angles made by $e(\ell)$ with its neighbor edges sharing the parent of ℓ in clockwise and counterclockwise directions, respectively (see Figure A.2).

Here, we introduce the following property.

Property $(*)$: Both $a_R(\ell) \neq \pi$ and $a_L(\ell) \neq \pi$.

If a leaf ℓ of T has Property $(*)$, the neighbor points on $\partial(P)$ are vertices of P. That is, for the parent u of ℓ, we have an edge (or straight line) $u'u''$ on $\partial(P)$ such that u' and u'' are vertices of P and are glued together to make the vertex u on Q, and the midpoint of the line segment $u'u''$ in $\partial(P)$ corresponds to ℓ on T. In other words, when the curvatures at u' and u'' are both unequal to π, it is easy to find the corresponding vertex ℓ of T, or a

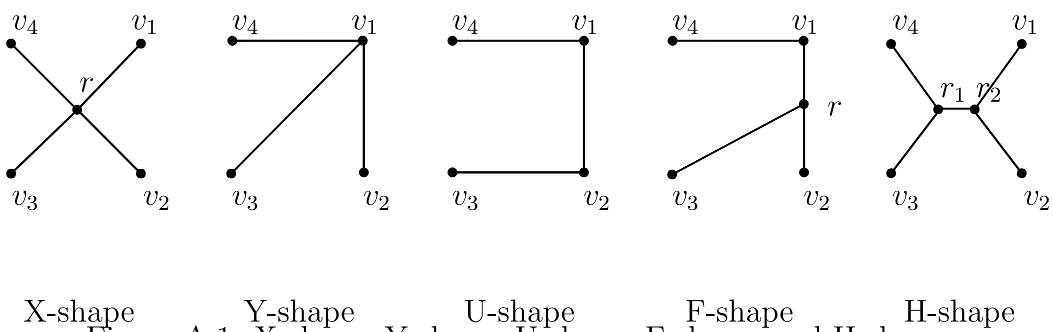

Figure A.1: X-shape, Y-shape, U-shape, F-shape, and H-shape.

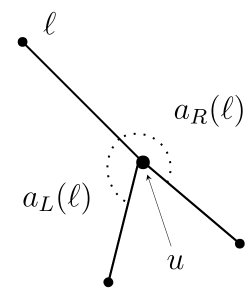

Figure A.2: Two angles $a_R(\ell)$ and $a_L(\ell)$ for a leaf ℓ.

vertex v_i on Q of curvature π.

We classify the figure of the cutting trees of a tetramonohedron into Type 1, Type 2, or Type 3.

A.1 Type 1

In Type 1, T has at least two leaves that satisfy Property $(*)$. Let ℓ be a leaf satisfying Property $(*)$ and v be the parent of ℓ. It is then easy to see that there is a line segment made from two copies of the edge $v\ell$ of T on $\partial(P)$. In other words, $\partial(P)$ has two consecutive vertices p_i and p_{i+1} such that the midpoint of the line segment $p_i p_{i+1}$ will be folded into ℓ, which is one of among v_1, v_2, v_3, and v_4, and p_i and p_{i+1} are glued together to make the vertex v on Q. Because T has two leaves, we can obtain two of among v_1, v_2, v_3 and v_4. Thus, we obtain Algorithm 5.

Algorithm 5: Folding algorithm for Type 1

Input : A polygon $P = (p_0, p_1, \ldots, p_{n-1}, p_0)$
Output: All possible pairs (p, p') of two points of P that correspond to vertices of a tetramonohedron in Type 1(if one exists)

```
foreach pair of two edges {e, e'} of P do
    take the midpoints m of e and m' of e';
    Output: (p, p') = (m, m')
```

A.2 Type 2

In Type 2, the set of cut lines of Q contains two independent line segments, say, v_1v_2 and v_3v_4 as in Figure A.3. When T is Type 2, we cut along these lines and obtain a cylinder, which is called a "rolling belt" in [9]. After that, the cylinder is cut and unfolded, and P is obtained. Therefore, there are two edges e and e' in $\partial(P)$ corresponding to v_1v_2 and v_3v_4. Inversely, when we glue the boundary of P except e, e', we obtain a cylinder. We can obtain Q from the cylinder by gluing their boundary suitably in infinitely many distinct ways. These ways contain the case that e is folded in half. Therefore, it is sufficient to check only the case where e is folded in half. In this case, the midpoint of e is corresponded to a vertex of a tetramonohedron, two end points of e are glued, and another vertex of a tetramonohedron is made. Thus, we obtain Algorithm 6.

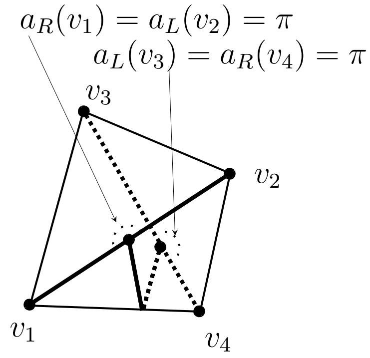

Figure A.3: Each leaf v_i satisfies $a_R(v_i) = \pi$ or $a_L(v_i) = \pi$ in Type 2.

Algorithm 6: Folding algorithm for Type 2

Input : A polygon $P = (p_0, p_1, \ldots, p_{n-1}, p_0)$

Output: All possible pairs (p, p') of two points of P that correspond to vertices of a tetramonohedron in Type 2(if one exists)

foreach ***edge e of P*** **do**

  take the midpoints m of e and an end point s of e.

    Output: $(p, p') = (m, s)$

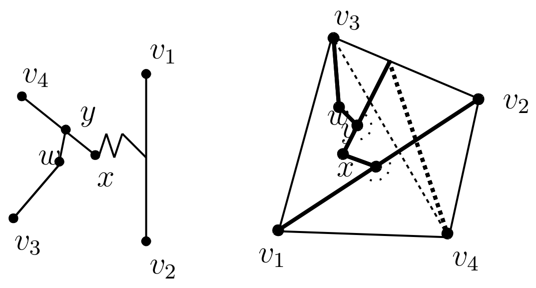

Figure A.4: Type 3: Only v_3 satisfies Condition $(*)$.

A.3 Type 3

In Type 3, the set of cut lines of Q contains two independent line segments, say, v_1v_2 and xv_4 with $x \neq v_3$. In this case, only v_3 satisfies Property $(*)$. For T, we have the situation shown in Figure A.4. Here, v_1 and v_2 are joined by a straight line and there are three vertices x, y, w such that v_4 and x are joined by a straight line, y is on the line segment v_4y, and v_3 is joined to y with some cut lines. (Note that v_3 can be w.) Therefore, for $\partial(P)$, v_3 is the midpoint of an edge of P, we have the same sequence of length from v_3 to both sides, and when we find the first pair e and e' of $|e| > |e'|$, then $e' = (w, y')$ and $e = (y'', v_4, y''', x)$, where y', y'', y''' are the three vertices forming y on Q. Hence, we can find point v_4 on $\partial(P)$ uniquely. Thus, we can determine if P can fold onto a tetramonohedron Q in Type 3 by Algorithm 7.

Algorithm 7: Folding algorithm for Type 3

Input : A polygon $P = (p_0, p_1, \ldots, p_{n-1}, p_0)$
Output: All possible pairs (p, p') of two points of P that correspond to vertices of a tetramonohedron in Type 3(if one exists)
foreach ***an edge e of P*** **do**
- Take the midpoint m of e as v_3
- Trace the same sequence of length from v_3 to both sides until we find the first pair e and e' of $|e| > |e'|$.
- Take the midpoint m' of e'.
- **Output:** $(p, p') = (m, m')$

A.4 Characterization

For a given polygon P, we apply three Algorithms 5, 6, and 7 individually. It takes $O(n^2)$ time.

After that, we show that any cut lines of Q to obtain P should be one of these types. Before the proof, we give a technical lemma for Property $(*)$.

Lemma 41 *1. For any leaf ℓ, it satisfies Property $(*)$ if its parent v has degree 2.*

2. If two leaves ℓ and ℓ' share their parent v of degree 3 and the angle between $e(\ell)$ and $e(\ell')$ is not π, then at least one of ℓ and ℓ' satisfies Property $(*)$.

3. When four leaves share their parent r, at least two leaves satisfy Property $(*)$.

Proof. (1) If the parent v is a vertex of Q, we have $a_R(\ell) + a_L(\ell) = \pi$ and $0 < a_R(\ell), a_L(\ell) < \pi$. Thus, Property $(*)$ holds.
(2) Because ℓ and ℓ' share an angle, $a_R(\ell) = a_L(\ell')$ or $a_R(\ell') = a_L(\ell)$. Without loss of generality, we assume that $a_R(\ell) = a_L(\ell')$. Then, $a_R(\ell) \neq \pi$ based on the assumption. Because $a_R(\ell)+a_R(\ell')+a_L(\ell) = 2\pi$, we have $a_R(\ell')+a_L(\ell) \neq \pi$. Therefore, at least one of $a_R(\ell')$ and $a_L(\ell)$ is not equal to π. Thus, at least one of ℓ and ℓ' satisfies Property $(*)$.
(3) Because r has four child leaves, one angle at most can be equal to or greater than π, and the other three angles are consecutively less than π. Let ℓ and ℓ' be the leaves between these three consecutive angles of less than π. Then, these two leaves satisfy Property $(*)$. □

Lemma 42 *Any cut lines of a tetramonohedron is one of either Type 1, Type 2, or Type 3.*

Proof. We show that most cases are of Type 1 except for two special cases, which imply Types 2 and 3.

X-shape: Here v_1, v_2, v_3 and v_4 are all leaves, and there is a vertex r in T with $\deg(r) = 4$. By Lemma 3(3), this case is of Type 1.

Y-shape: Three of among v_1, v_2, v_3 and v_4 are leaves, and the last one is a vertex of degree 3. Without loss of generality, we assume that $\deg(v_1) = 3$ and $\deg(v_2) = \deg(v_3) = \deg(v_4) = 1$. Then, none of the three angles $\angle v_2v_1v_3$, $\angle v_3v_1v_4$, and $\angle v_4v_1v_2$ is equal to π because v_1 is a vertex of Q of a curvature $\angle v_2v_1v_3+\angle v_3v_1v_4+\angle v_4v_1v_2 = \pi$, and $0 < \angle v_2v_1v_3, \angle v_3v_1v_4, \angle v_4v_1v_2$. Hence, this case is of Type 1.

U-shape: Two of among v_1, v_2, v_3 and v_4 are leaves, and the other two are vertices of degree 2. Without loss of generality, we assume that $\deg(v_1) = \deg(v_2) = 2$ and $\deg(v_3) = \deg(v_4) = 1$, and v_1 is closer to v_4 than v_2. If T has a vertex u of degree 2 between v_1 and v_4, v_4 has Property $(*)$ by Lemma 3(1). Thus, we consider the case in which v_1 is the parent of v_4. However, v_4 has Property $(*)$ by Lemma 3(1) for the parent v_1. The leaf v_3 also satisfies Property $(*)$ by the same argument with v_2. Thus, this case is of Type 1.

F-shape: Three of among v_1, v_2, v_3 and v_4 are leaves, and the last one is a vertex of degree 2, and T has another vertex r of $\deg(r) = 3$. Without loss of generality, v_1 is the vertex of degree 2, and v_2 is the leaf reachable to v_1 without going through r. Then, by the same argument of the U-shape, v_2 satisfies Property $(*)$. In the same way, if v_3 or v_4 has other vertices of degree 2 on the way to r, it satisfies Property $(*)$. Thus, we consider the other case in which both v_3 and v_4 are children of r. If the angle $\angle v_3 r v_4 \neq \pi$, by Lemma 3(2), one of v_3 and v_4 satisfies Property $(*)$. On the other hand, when $\angle v_3 r v_4 = \pi$, we have two independent line segments $v_1 v_2$ and $v_3 v_4$. This case is of Type 2. Intuitively, the cut lines $v_1 v_2$ and $v_3 v_4$ open Q into a cylinder, and the cylinder is open by cutting the line segment(s) joining v_1 and r.

H-shape: Here, v_1, v_2, v_3 and v_4 are all leaves, and there are two vertices r_1 and r_2 in T with $\deg(r_1) = \deg(r_2) = 3$. We assume that r_1 has children v_1 and v_2, and r_2 has children v_3 and v_4. (When the other vertices are between them, we can reduce to the other cases above.) If $\angle v_1 r_1 v_2 \neq \pi$ and $\angle v_3 r_2 v_4 \neq \pi$, by Lemma 3(2), two vertices r_1 and r_2 have at least one leaf satisfying Property $(*)$. By contrast, when $\angle v_1 r_1 v_2 = \pi$ and $\angle v_3 r_2 v_4 = \pi$, we have the case of Type 2. For the last case, without loss of generality, $\angle v_1 r_1 v_2 = \pi$ and $\angle v_3 r_2 v_4 \neq \pi$. Let r' be the third neighbor of r_2 other than v_3 and v_4 (which can be $r' = r_2$). If $\angle r' r_2 v_3 \neq \pi$ and $\angle r' r_2 v_4 \neq \pi$, both v_3 and v_4 satisfy Property $(*)$, which is of Type 1. Type 3 is the last remaining case in which $\angle v_1 r_1 v_2 = \pi$ and $\angle v_4 r_2 r' = \pi$.

Because all cases are covered by the results in [2], we can conclude that three Algorithms 5, 6, and 7 can determine whether P can fold onto Q. □

Publications

Journal Papers

- Tonan Kamata, Akira Kadoguchi, Takashi Horiyama, and Ryuhei Uehara. Efficient Folding Algorithms for Convex Polyhedra. *Discrete and Computational Geometry*, 24 pages, accepted, January 2022. DOI:10.1007/s00454-022-00415-7.

- Erik D. Demaine, Martin L. Demaine, Yevhenii Diomidov, Tonan Kamata, Ryuhei Uehara, and Hanyu Alice Zhang. Any Regular Polyhedron Can Transform to Another by O(1) Refoldings. *Computational Geometry: Theory and Applications*, (21 pages), submitted, January 2022.

Conference Papers

- Tonan Kamata, Akira Kadoguchi, Takashi Horiyama, and Ryuhei Uehara. Efficient Folding Algorithms for Regular Polyhedra. *The 32nd Canadian Conference on Computational Geometry*, pp. 121–127, 2020.

- Erik D. Demaine, Martin L. Demaine, Yevhenii Diomidov, Tonan Kamata, Ryuhei Uehara, and Hanyu Alice Zhang. Any regular polyhedron can transform to another by O(1) refoldings. *The 33rd Canadian Conference on Computational Geometry*, pp. 332–342, 2021.

- Elena Arseneva, Erik D. Demaine, Tonan Kamata, and Ryuhei Uehara. Discretization to Prove the Nonexistence of "Small" Common Unfoldings Between Polyhedra. *The 34th Canadian Conference on Computational Geometry*, pp. 9–15, 2022.

References

[1] Jin Akiyama. Tile-makers and semi-tile-makers. *American Mathematical Monthly*, 114(7):602–609, 2007.

[2] Jin Akiyama and Kiyoko Matsunaga. An algorithm for folding a conway tile into an isotetrahedron or a rectangle dihedron. *Journal of Information Processing*, 28:750–758, 2020.

[3] Alexandr D. Alexandrov. Existence of a convex polyhedron and a convex surface with a given metric. In *A. D. Alexandrov: Selected Works*, volume Part I, pages 169–173. Gordon and Breach, Australia, 1996.

[4] Nathan Altshiller-Court. *Modern Pure Solid Geometry.* Chelsea, 1979.

[5] Yoshiaki Araki, Takashi Horiyama, and Ryuhei Uehara. Common unfolding of regular tetrahedron and johnson-zalgaller solid. *Journal of Graph Algorithms and Applications*, 20(1):101–114, 2016.

[6] Amartya Shankha Biswas and Erik D. Demaine. Common development of prisms, anti-prisms, tetrahedra, and wedges. In *the 29th Canadian Conference on Computational Geometry*, pages 202–207, 2017.

[7] Mark de Berg, Otfried Cheong, Mark van Kreveld, and Mark Overmars. *Computational Geometry Algorithms and Applications.* Springer-Verlag, 3 edition, 2008.

[8] Erik D. Demaine, Martin L. Demaine, Jin-ichi Itoh, Anna Lubiw, Chie Nara, and Joseph O'Rourke. Refold rigidity of convex polyhedra. *Computational Geometry: Theory and Applications*, 46(8):979–989, 2013.

[9] Erik D. Demaine and Joseph O'Rourke. *Geometric folding algorithms - linkages, origami, polyhedra.* Cambridge University Press, 2007.

[10] Albrecht Dürer. Underweysung der messung, mit den zirckel un richtscheyt, in linien ebnen unnd gantzen corporen, 1525.

[11] Takashi Horiyama and Koichi Mizunashi. Folding Orthogonal Polygons into Rectangular Boxes. In *the 19th Japan-Korea Joint Workshop on Algorithms and Computation (WAAC 2016)*, 2016.

[12] Takashi Horiyama and Ryuhei Uehara. Nonexistence of common edge developments of regular tetrahedron and other platonic solids. In *China-Japan Joint Conference on Computational Geometry, Graphs and Applications (CGGA 2010)*, pages 56–57, 2010.

[13] Jun Mitani and Ryuhei Uehara. Polygons folding to plural incongruent orthogonal boxes. In *the 20th Annual Canadian Conference on Computational Geometry*, pages 39–42, 2008.

[14] Koichi Mizunashi, Takashi Horiyama, and Ryuhei Uehara. Efficient algorithm for box folding. *Journal of Graph Algorithms and Applications*, 24(2):89–103, 2020.

[15] Dawei Xu, Takashi Horiyama, Toshihiro Shirakawa, and Ryuhei Uehara. Common Developments of Three Incongruent Boxes of Area 30. *Computational Geometry: Theory and Applications*, 64:1–17, 2017.