This advance publication article will be replaced by the finalized version after proofreading.
PAPER

Overlapping of Lattice Unfolding for Cuboids*

Takumi SHIOTA†, Student Member, Tonan KAMATA††, Nonmember, and Ryuhei UEHARA†††, Member

SUMMARY A polygon obtained by cutting the surface of a polyhedron is called an unfolding. An unfolding obtained by cutting along only edges is called an edge unfolding. An unfolding may have overlapping, which are self-intersections on its boundary. It is a well-known open question in computational origami whether or not every convex polyhedron has a non-overlapping edge unfolding. On the other hand, Sharir and Schorr showed that any convex polyhedron could unfold without overlapping when allowed to cut its faces. Therefore, there is a gap between edge unfoldings and general unfoldings. Bridging this gap is necessary as a foothold on this open question of edge unfolding. Instead of cutting faces arbitrarily, there are studies considering whether specific cutting lines on the faces can result in unfoldings without overlaps. Lattice unfoldings of a cuboid made by unit cubes are one such example. A lattice unfolding of a cuboid is a polygon obtained by cutting the faces along the edges of unit squares. An unfolding may have overlapping, even in the case of small cuboids. In particular, Uno showed that a $1 \times 1 \times 3$-cuboid has an overlapping lattice unfolding, while Mitani and Uehara showed the same for three faces of a $1 \times 2 \times 3$-cuboid. In contrast, it is known that some cuboids have no overlapping lattice unfolding. Hearn showed it for a $1 \times 1 \times 2$-cuboid, and Sugihara showed the same for a $2 \times 2 \times 2$-cuboid. In this study, we completely clarify the existence of overlapping lattice unfoldings, which also contains the case where the sizes are non-integers.

key words: computational origami, polyhedron, overlapping unfolding, cuboid, lattice unfolding

1. Introduction

To represent a polyhedron, we sometimes use a planer layout of arranged faces according to their adjacency relations. The origin of this method can be traced back to Albrecht Dürer’s 1525 book “Underweysung der messung mit dem zirkel un richt scheyt” [1]. He represented several polyhedra using flat polygons (edge unfoldings) obtained by cutting along the edges. All edge unfoldings of convex polyhedra in this book are drawn so that “no two faces overlap.” However, edge unfoldings of polyhedra do not always satisfy this condition (e.g., Namiki and Fukuda’s overlapping edge unfolding as shown in Fig. 1). The following problem is open:

Open Problem 1 ([2], Open Problem 21.1). Does every convex polyhedron have a non-overlapping edge unfolding?

Fig. 1: An overlapping edge unfolding of a cube with cut-off corners [3]. Cut along thick lines to get the figure on the right.

Fig. 2: An overlapping lattice unfolding in the $1 \times 1 \times 3$-cuboid

Research on the existence of unfolding with overlap for polyhedra has been conducted under several different conditions. Biedl et al. discovered concave polyhedra where all edge unfoldings overlap in 1998, and Grünbaum found another instance in 2003 [4], [5]. For convex regular-faced polyhedra, which are polyhedra whose faces are all regular polygons, it has been completely determined whether they have overlapping edge unfoldings [6]–[9]. Additionally, the number of non-overlapping edge unfoldings has been counted for the convex regular-faced polyhedra with overlapping edge unfoldings [10].

There are also studies on general unfoldings that allow cutting the faces of the polyhedron, not just its edges. Sharir and Schorr showed that any convex polyhedron could unfold without overlapping when allowed to cut its faces [11], [12]. Therefore, there is a gap between edge unfoldings and general unfoldings. Bridging this gap is necessary as a foothold on Open Problem 1. Instead of cutting faces arbitrarily, there are studies considering whether specific cutting lines on the faces can result in unfoldings without overlaps. Lattice unfoldings of a cuboid are one such example.

In 2008, Uno showed that the $1 \times 1 \times 3$-cuboid has an overlapping lattice unfolding (Fig. 2) [13]. Furthermore, in 2008, Mitani and Uehara showed that the $1 \times 2 \times 3$-cuboid has an overlapping lattice unfolding (Fig. 3) [14].

Each of these cutting methods can be extended to the $1 \times 1 \times z$-cuboid, where $z \geq 3$ and the $1 \times y \times z$-cuboid,
where \( y \geq 2, z \geq 3 \), respectively. The following theorems are obtained:

**Theorem 2 ([13]).** The \( 1 \times 1 \times z \)-cuboid, where \( z \in \mathbb{N} \) and \( z \geq 3 \), has an overlapping lattice unfolding.

**Theorem 3 ([14]).** The \( 1 \times y \times z \)-cuboid, where \( y, z \in \mathbb{N}, y \geq 2, z \geq 3 \), and \( y \leq z \), has an overlapping lattice unfolding.

On the other hand, the following results are known for the non-existence of overlapping lattice unfolding:

**Theorem 4 ([15]).** The \( 1 \times 1 \times 2 \)-cuboid has no overlapping lattice unfolding.

**Theorem 5 ([16]).** The \( 2 \times 2 \times 2 \)-cuboid has no overlapping lattice unfolding.

Cutting lines can be taken not only parallel to the edges of the cuboid but also diagonally (Fig. 4). Furthermore, as shown in Fig. 5, it is known that in the edge unfoldings of polyhedra, two edges or two vertices of faces can be in contact [9]. In this study, we successfully clarified the sizes at which the lattice unfoldings of all cuboid including those with diagonal lattice cutting lines, overlap. These overlaps include three types: faces overlapping, two edges in touch, and two vertices in touch. This result contributes to bridging the gap between edge unfoldings and general unfoldings.

## 2. Preliminaries

### 2.1 Definition of cuboids

Let’s consider a square lattice where each square has an area of \( 1 \times 1 \). Suppose \( A = (a, 0) \) and \( B = (0, b) \) are a pair of lattice points, where \( a \in \mathbb{N}^+, b \in \mathbb{N}, a \geq b \) (Fig. 6). Consider a square with a side \( AB \), whose length is \( L = \sqrt{a^2 + b^2} \). A **cube with length \( L \) on a side** is constructed by assembling the squares as its faces (Fig. 7(a)).

A \((xL, yL, zL)\)-cuboid is defined as a box with edge lengths \( xL, yL, \) and \( zL \) along the \( x \)-axis, the \( y \)-axis, and the \( z \)-axis, respectively, for some positive integers \( x, y, \) and \( z \) (Fig. 7(b)). Here, \( x \leq y \leq z \) is assumed without loss of gen-
Lemma 6 (See e.g., [2], Lemma 22.1.1). A subgraph $G \subseteq G_Q$ generates an unfolding if and only if $G$ is a spanning tree of $G_Q$.

2.2 Definition of overlapping lattice unfoldings

A lattice unfolding of a cuboid is a planar shape obtained by cutting the face of the cuboid along the edges of unit squares (Fig. 7(c)). As we will mention in Lemma 7, the cutting line of the lattice unfolding forms a tree structure.

On a lattice unfolding, the original cuboid’s unit squares are arranged planarly so that their edges are glued together. Any pair of unit squares not adjacent on the surface can be classified into positional relationships as follows:

1. Overlap in the same position (Fig. 9(a)).
2. Share one edge (Fig. 9(b)).
3. Share one vertex (Fig. 9(c)).
4. Without sharing any edges or vertices.

Herein, we say that an unfolding is faces-in-touch if it has a pair of unit squares satisfying (1). Similarly, we define edges-in-touch and vertices-in-touch for (2) and (3), respectively. When all pairs of unit squares not adjacent on the surface satisfy (4), it is called non-overlapping. When any of the conditions (1), (2), or (3) are satisfied, it is termed overlapping. Note that the inclusion relationship \{faces-in-touch unfoldings\} $\subseteq$ \{edges-in-touch unfoldings\} $\subseteq$ \{vertices-in-touch unfoldings\} holds for any cuboid.

2.3 Representation of polyhedra using graphs

Let $Q$ be a polyhedron, and $G_Q = (V_Q, E_Q)$ be the graph such that $V_Q$ is the set of vertices of $Q$ and $E_Q$ is the set of the edges of $Q$. We call this graph an edge representation graph of $Q$. An edge unfolding of $Q$ can be regarded as an unfolding obtained from a subgraph of $G_Q$. The following lemma holds:

Lemma 7 (see Fig. 10; [14], Theorem 1, Theorem 3). Let $G_C = (V_C, E_C)$ be the lattice representation graph of a cuboid $C$, and let $S(V_C) \subseteq V_C$ be the set of lattice points located at the vertices of $C$. Then, the following are equivalent for a subgraph $G \subseteq G_C$:

1. $G$ yields a lattice unfolding.
2. $G$ is a tree that satisfies $S(V_C) \subseteq G$, and for any vertex $v$ in $G$, if $\deg(v) = 1$, then $v \in S(V_C)$ (where $\deg(v)$ is the degree of vertex $v$).
Fig. 11: Pairs of faces to check for overlap in an edge unfolding of a (1,1,1)-cuboid

(a): All pairs (b): Minimum pairs

Fig. 12: (a) An edge unfolding in a cube cut-off corners (Fig. 1). (b) (a)’s MOPE. Removing any face results in non-connected structures, contradicting the definition of partial edge unfoldings. (c) Removing the gray face results in MOPE.

2.4 Methods for checking the overlap

Herein, we introduce a method for verifying the non-existence of overlapping edge unfoldings for polyhedron $Q$.

To show the non-existence of overlapping edge unfolding in $Q$, we check the overlapping for all pairs of faces of all edge unfoldings. For example, a $(1, 1, 1)$-cuboid’s edge unfolding has $sC_2 = 15$ pairs of squares that need to be checked for overlap (see Figure 11(a)). On the other hand, focusing on the symmetry of relative positions, the number of pairs that actually need to be checked is six (see Fig. 11(b)). In other words, if we check that none of them overlap, we can conclude that all edge unfoldings do not overlap.

An algorithm called rotational unfolding has been developed with a focus on this point [9]. Herein, let polyhedron $Q$ be as a dual graph $D(G_Q) = (V_D, E_D)$, where $V_D$ is a set of faces in the polyhedron, and $E_D$ is a set of edges such that two vertices are adjacent if and only if the corresponding two faces are neighbors. Let a partial edge unfolding be a flat polygon formed from a set of faces corresponding to a connected induced subgraph of $D(G_Q)$. Rotational unfolding can enumerate minimal overlapping partial edge unfoldings (MOPEs), partial edge unfoldings with the minimal faces required to connect two overlapping faces. Figure 12 shows an example of a MOPE and a non-MOPE. Also, partial edge unfoldings in Fig. 11(b) are MOPEs. In rotational unfolding, each MOPE is enumerated by “rolling the polyhedron on a plane from the state that one face is bottom to the state that another is bottom.” For details on rotational unfolding, refer to [9].

3. Results

This study presents the following theorem for cuboids:

**Theorem 8.**

- Both the $(1, 1, 1)$-cuboid and $(\sqrt{2}, \sqrt{2}, \sqrt{2})$-cuboid have no overlapping lattice unfolding.
- The $(1, 1, 2)$-cuboid has neither faces-in-touch lattice unfolding nor edges-in-touch lattice unfolding, but it has a vertices-in-touch lattice unfolding.
- Both the $(1, 2, 2)$-cuboid and $(2, 2, 2)$-cuboid have no faces-in-touch lattice unfolding, but they have edges-in-touch lattice unfolding and vertices-in-touch lattice unfolding.
- Any other type of cuboids have faces-in-touch lattice unfoldings, edges-in-touch lattice unfoldings, and vertices-in-touch lattice unfoldings.

Hereafter, we explain the non-existence side of this Theorem in Section 3.1 and the existence side in Section 3.2.

3.1 The method to check the non-existence of overlapping lattice unfoldings by computational experiment

First, we show a method to check the non-existence of overlapping lattice unfoldings through a computational experiment using rotational unfolding described in Section 2.4. However, using it directly for lattice unfolding is inefficient for the search. In this section, we present the method of extending rotational unfolding to lattice unfolding and the results of computational experiments.

In the rotational unfolding for polyhedron $Q$, the dual graph $D(G_Q)$ of its edge representation graph $G_Q$ is used. Accordingly, we consider the dual graph $D(G_C)$ of the lattice representation graph $G_C$ for the lattice unfolding of a cuboid $C$. However, using rotation unfolding directly for $D(G_C)$ results in including partial lattice unfoldings that are not minimal overlapping partial lattice unfoldings (in short, MOPL; see example in Fig. 13). Including partial lattice unfoldings that are non-MOPLs reduces efficiency when
checking for the existence of overlapping lattice unfoldings. Here, we introduce the following characters for information about the "direction of rolling when viewed from one step before":

R: Roll to the right from one step before.
C: Roll straight from one step before.
L: Roll to the left from one step before.

Therefore, the partial lattice unfolding obtained directly using the rotational unfolding can be represented as a string (see example in Fig. 14). In the rotational unfolding, the first step is to roll straight ahead without loss of generality, so the string corresponding to the partial lattice unfolding obtained in the first step is “C”. Here, we can show the following lemma:

**Lemma 9.** When the strings corresponding to the partial lattice unfoldings include “RR” or “LL”, they are non-MOPLs.

**Proof.** In the second step of the rotational unfolding, we have three cases: (1) rolling to the right (“CR”; Fig. 15(a)), (2) rolling straight (“CC”; Fig. 15(b)), and (3) rolling to the left (“CL”; Fig. 15(c)). If we repeat the action of rolling right, or “RR”, twice after the second step, we get (1) “CRRR” (Fig. 15(d)), (2) “CCRR” (Fig. 15(e)), and (3) “CLRR” (Fig. 15(f)). For case (1), this situation cannot occur because we have already used the face as part of the constructed partial edge unfolding. For cases (2) and (3) (Fig. 15(e) and Fig. 15(f)), these partial lattice unfoldings are non-MOPLs, and removing the plaid faces results in MOPLs Fig. 15(a) and Fig. 15(b). The same statement applies even if “RR” appears not only in the first four steps but also at any point during the rolling process. Similarly, the same can be said for “LL”.

Therefore, if “RR” or “LL” appears during rolling, it is a non-MOPL; there is no need to continue rolling, thereby pruning the search.

When a cuboid has an overlapping lattice unfolding, we can determine how they overlap using the following observation:

**Observation 10.** In rotational unfolding, compute the center coordinates of the face at one endpoint, assuming its center coordinates are (0, 0) (see Fig. 16(a)). Then, while rolling the cuboid sequentially, compute the center coordinates of the face at the other endpoint in the partial lattice unfolding. We can determine the type of unfolding based on the coordinates of the center of the face at the other endpoint:

- If the coordinate is (0, 0), it is a faces-in-touch unfolding (a plaid face in Fig. 16(b)).
- If the coordinates are (0, 1), (−1, 0), or (0, −1), it is an edges-in-touching unfolding (polka dot faces in Fig. 16(b)).
- If the coordinates are (1, 1), (1, −1), (−1, 1), or (−1, −1), it is a vertices-in-touch unfolding (striped faces in Fig. 16(b)).

We implemented the method of extending rotational unfolding to lattice unfolding and obtained the non-existence results shown in Theorem 8. Experiments were conducted on a Mac OS Venture computer with an Apple M1 Max chip and 64GB of memory. Tables 1 to 3 show the running times of computational experiments for each lattice cuboid. These experiment results include verifying the previous results [15] and [16].

3.2 Proving the existence of overlapping lattice unfoldings by constructing specific examples

Hereafter, we prove the existence side of the statements of Theorem 8 by showing specific overlapping unfoldings.
Table 1: The running time to demonstrate the non-existence of faces-in-touch unfoldings.

<table>
<thead>
<tr>
<th>Lattice cuboid</th>
<th># Faces</th>
<th># Edges</th>
<th># Vertices</th>
<th>Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>(1, 1, 1)-cuboid</td>
<td>6</td>
<td>12</td>
<td>8</td>
<td>0.3s</td>
</tr>
<tr>
<td>(1, 1, 2)-cuboid</td>
<td>10</td>
<td>20</td>
<td>12</td>
<td>0.6s</td>
</tr>
<tr>
<td>(1, 2, 2)-cuboid</td>
<td>16</td>
<td>32</td>
<td>18</td>
<td>1.6s</td>
</tr>
<tr>
<td>(√2, √2, √2)-cuboid</td>
<td>24</td>
<td>48</td>
<td>26</td>
<td>56.9s</td>
</tr>
<tr>
<td>(√2, √2, √2)-cuboid</td>
<td>12</td>
<td>24</td>
<td>14</td>
<td>0.7s</td>
</tr>
</tbody>
</table>

Table 2: The running time to demonstrate the non-existence of edges-in-touch unfoldings.

<table>
<thead>
<tr>
<th>Lattice cuboid</th>
<th># Faces</th>
<th># Edges</th>
<th># Vertices</th>
<th>Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>(1, 1, 1)-cuboid</td>
<td>6</td>
<td>12</td>
<td>8</td>
<td>0.3s</td>
</tr>
<tr>
<td>(1, 1, 2)-cuboid</td>
<td>10</td>
<td>20</td>
<td>12</td>
<td>0.5s</td>
</tr>
<tr>
<td>(√2, √2, √2)-cuboid</td>
<td>12</td>
<td>24</td>
<td>14</td>
<td>0.7s</td>
</tr>
</tbody>
</table>

Table 3: The running time to demonstrate the non-existence of vertices-in-touch unfoldings.

<table>
<thead>
<tr>
<th>Lattice cuboid</th>
<th># Faces</th>
<th># Edges</th>
<th># Vertices</th>
<th>Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>(1, 1, 1)-cuboid</td>
<td>6</td>
<td>12</td>
<td>8</td>
<td>0.4s</td>
</tr>
<tr>
<td>(√2, √2, √2)-cuboid</td>
<td>12</td>
<td>24</td>
<td>14</td>
<td>0.7s</td>
</tr>
</tbody>
</table>

3.2.1 Case of \(L = 1\)

From Theorems 2 and 3, faces-in-touch, edges-in-touch, and vertices-in-touch unfoldings exist for the \((x, y, z)\)-cuboid, where \(z \geq 3\). For the remaining cases for the case of \(L = 1\), we provide specific examples of unfoldings as follows:

**Lemma 11.**

- The \((1, 1, 2)\)-cuboid has a vertices-in-touch unfolding (Fig. 17(a)).
- The \((1, 2, 2)\)-cuboid has both an edges-in-touch unfolding (Fig. 17(b)) and a vertices-in-touch unfolding (Fig. 17(c)).
- The \((2, 2, 2)\)-cuboid has both an edges-in-touch unfolding (Fig. 17(d)) and a vertices-in-touch unfolding (Fig. 17(e)).

3.2.2 Case of \(L = \sqrt{2}, L = \sqrt{5}, \text{ and } L = \sqrt{10}\)

From the inclusion relationship between the edges-in-touch and vertices-in-touch unfolding, we have only to show the existence of the faces-in-touch unfolding.

A faces-in-touch unfolding exist for the \((\sqrt{2}, \sqrt{2}, 2\sqrt{2})\)-cuboid (Fig. 17(f)). From now on, we will refer to this partial lattice unfolding as \(Q_L\) (Fig. 18). Moreover, the \((\sqrt{2}, \sqrt{2}, 2\sqrt{2})\)-cuboid can be unfolded to include the partial lattice unfolding \(Q_L\) because \(Q_L\) can be embedded in the three faces in front of the \((\sqrt{2}, \sqrt{2}, 2\sqrt{2})\)-cuboid (see Fig. 19).

Note that we have to fold the three triangular faces: a plaid face in the positive \(y\)-axis direction, a polka dot face in the positive \(x\)-axis direction, and a striped face in the positive \(x\)-axis direction. This embedding method can also be applied to the \((x\sqrt{2}, y\sqrt{2}, z\sqrt{2})\)-cuboid, where \(x, y, z \geq 2\), as shown in Fig. 20.

The same embedding can be performed for cases where \(L = \sqrt{5}\) and \(L = \sqrt{10}\) (see Fig. 21(a) and Fig. 21(b)).

3.2.3 Case of \(L \geq \sqrt{13}\)

The partial lattice unfolding \(Q_L\) can be embedded in the \((\sqrt{13}, \sqrt{13}, \sqrt{13})\)-cuboid, as shown in Fig. 21(c). Here, we present the following lemma:
Fig. 19: $Q_L$ can be embedded in the three faces in front of the $(\sqrt{2}, \sqrt{2}, 2\sqrt{2})$-cuboid.

Fig. 20: $Q_L$ can be embedded in the $(x\sqrt{2}, y\sqrt{2}, z\sqrt{2})$-cuboid, where $z \geq 2$.

(a): The $(\sqrt{3}, \sqrt{3}, \sqrt{3})$-cuboid
(b): The $(\sqrt{10}, \sqrt{10}, \sqrt{10})$-cuboid
(c): The $(\sqrt{13}, \sqrt{13}, \sqrt{13})$-cuboid
(d): The $(L, L, L)$-cuboid, where $L \geq \sqrt{13}$

Fig. 21: $Q_L$ can be embedded in each cuboid.

Lemma 12. The partial lattice unfolding $Q_L$ can be embedded in the $(L, L, L)$-cuboid, where $L \geq \sqrt{13}$.

Proof. Consider the three unit squares with vertex $v$ in common (Fig. 21(d)). The three-unit squares enclosed in blue in Fig. 18 can be embedded in this point. Let $S$ be the side face of a cone with the length of axis $\sqrt{13}$ and a central angle of 270° (Fig. 22). Hereafter, $S$ is called the cone. Since the central angle of the cone $S$ is 270°, the three unit squares enclosed in blue in Fig. 18 can be embedded with vertex $v$ coinciding. Additionally, due to the Euclidean distance between the point $v$ and the furthest point $w$ in Fig. 18 being $\sqrt{2^2 + 3^2} = \sqrt{13}$, the remaining faces, except for the three faces enclosed in blue, can be embedded as shown in Fig. 22 (right). The cone $S$ can be embedded in the three front faces of a $(L, L, L)$-cuboid where $L \geq \sqrt{13}$, as shown in Fig. 23. From the fact that the cone $S$ can be embedded in a $(L, L, L)$-cuboid and that $Q_L$ can be embedded on top of the cone $S$, we can concluded that $Q_L$ can be embedded in the three front faces of a $(L, L, L)$-cuboid.

From this lemma, a faces-in-touch unfolding exists for the $(xL, yL, zL)$-cuboid in any of the $x, y, z$, where $L \geq \sqrt{13}$. The same can be said for edges-in-touch and vertices-in-touch unfolding due to the inclusion relationship.

4. Conclusion

In this paper, we completely clarified the condition for a cuboid to have overlapping lattice unfoldings. This result is one approach to bridging the gap between edge unfoldings and general unfoldings. We can also extend the idea of considering lattice cutting lines on faces to the triangular lattice unfolding of an octahedron or icosahedron.

Furthermore, another approach to bridging this gap is to add cutting lines along the diagonals of the faces of convex regular-faced polyhedra. We believe that considering unfoldings with various discrete cutting lines on the faces could provide insights into solving Open Problem 1.

Acknowledgments.

This work was supported in part by JSPS KAKENHI Grant JP20H05961, 20H05964, 22H01423, and JP22KJ1480.
References


Takumi Shiota is a Ph.D. student at Kyushu Institute of Technology, and Research Fellow of Japan Society for the Promotion of Science. He received B.Eng. and M.Eng. degrees from Kyushu Institute of Technology in 2021 and 2023, respectively.

Tonan Kamata is an assistant professor at the School of Information Science, Japan Advanced Institute of Science and Technology (JAIST). He was a postdoctoral Fellow (PD) of a Japan Society for the Promotion of Science (JSPS) at JAIST. He received a B.S. from Tokyo University of Science, Japan, in 2019, and M.S. and Ph.D. degrees in Information Science from JAIST in 2021 and 2023, respectively.

Ryuhei Uehara is a professor at School of Information Science, Japan Advanced Institute of Science and Technology (JAIST). He received B.E., M.E., and Ph.D. degrees from University of Electro-Communications, Japan, in 1989, 1991, and 1998, respectively. He was a researcher in CANON Inc. during 1991–1993. In 1993, he joined Tokyo Woman’s Christian University as an assistant professor. He was a lecturer during 1998–2001, and an associate professor during 2001–2004 at Komazawa University. He moved to JAIST in 2004. His research interests include computational complexity, algorithms and data structures, and graph algorithms. Especially, he is engrossed in computational origami, games and puzzles from the viewpoints of theoretical computer science. He is a member of EATCS, and chair of EATCS Japan Chapter.