

Designing crease patterns for polyhedra by composing right frusta

Herng Yi Cheng, Kang Hao Cheong*

National University of Singapore High School of Mathematics and Science, 20 Clementi Avenue 1, 129957 Singapore, Singapore

ARTICLE INFO

Article history:

Received 9 June 2011

Accepted 18 November 2011

Keywords:

Geometry

Origami

Frustum

Crease pattern

Extrusion

ABSTRACT

We propose an original and novel algorithm for the automatic development of crease patterns for certain polyhedra with discrete rotational symmetry by composing right frusta. Unlike existing algorithms, the folded product will conform to the surface of the target polyhedron without external flaps. The crease patterns of frusta are drawn first and then composed by the algorithm to draw the crease pattern of the rotationally symmetric polyhedron. The composition is performed by splitting creases that were folded on pleats from frustum crease patterns. A CAD program has been written to implement the algorithm automatically, allowing users to specify a target polyhedron and generate a crease pattern that folds into it.

© 2011 Elsevier Ltd. All rights reserved.

1. Introduction

Origami, the art of paper-folding, has been the subject of extensive mathematical study. Demaine and O'Rourke define a folded origami as an isometric and noncrossing map from paper into Euclidean space [1]. In particular, much ground has been covered in the field of *origami design*, where algorithms are proposed to fold certain target shapes in two and three dimensions. Work includes Lang's generation of uniaxial base crease patterns via circle packing [1–3] and Tachi's tucking molecule algorithm to fold general polyhedral surfaces [4]. Demaine et al. have also derived algorithms to fold polygonal regions and polyhedra using rectangular strips; they have also generalized their algorithm to fold polygonal regions and polyhedra with two colors using bicolor paper [1,5]. Extrusion origami is the study of designing origami models that rise (“extrude”) from a piece of paper while the surrounding paper, despite minor shifts, is still flat. Much work has been done on this subject, including Benbernou et al.'s extrusion of polycubes [6] and Demaine et al.'s extrusion of orthogonal mazes [7].

Notably, Mitani has extruded solids with discrete rotational symmetry whose cross-sections normal to an axis are homothetic regular polygons, but whose outer surface may be curved or polyhedral [8,9]. The rotationally symmetric polyhedra investigated by Mitani are composed of right frusta, or the portion of a right pyramid between two cutting planes. His algorithm produces undesirable flaps outside the folded solid that cause the model to deviate

from the desired surface. Lang's design of rotationally symmetric origami pots [10] also faces the problem of external flaps (Fig. 1). While these flaps may not detract from the model aesthetically, they deviate from the intended form and restrict its potential applications. We propose an original and novel method to sequentially graft the crease pattern of each frustum so as to draw the crease patterns of rotationally symmetric polyhedra. This algorithm allows the extrusion of models such that the outer surface conforms to the target solid, without the external flaps. Another advantage is that the flat surrounding paper in the folded polyhedron can be folded into another origami model, extruding the polyhedron from a flat face of any origami model. Besides creating new artistic origami for industrial packaging, our method may also be applicable to nanofabrication in the future.

Section 2 provides the key definitions that will be used in the paper. Section 3 presents algorithms for drawing the crease patterns of right frusta and proves some of their properties. Section 4 derives the main result of this paper: the algorithm that composes right frustum crease patterns into the crease pattern of the rotationally symmetric polyhedron. Section 5 verifies the algorithm by folding the generated crease patterns of test examples specifically chosen for their varying structure. Section 6 discusses the implications of folding the polyhedra using bicolor paper and presents methods to ensure that the external surface of the folded product has only one color. Section 7 discusses some limitations and proposed future work.

2. Definitions

Definition 1 (Profile). A profile (Fig. 2(a)) is a non-self-intersecting, planar straight line path graph $G(V, E)$ embedded in \mathbb{R}^3 (but

* Corresponding author. Tel.: +65 65161587; fax: +65 67756910.

E-mail address: g0800484@nus.edu.sg (K.H. Cheong).

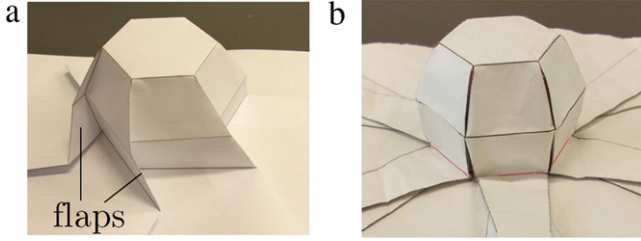


Fig. 1. The same solid folded by (a) Mitani's algorithm with external flaps and (b) our proposed algorithm without external flaps.

restricted to the xz plane) with vertices and edges labeled as follows:

$$V = \{v_0, v_1, v_2, \dots, v_m\} \quad \text{and} \quad (2.1)$$

$$E = \{e_1, e_2, \dots, e_m\} \quad \text{where } e_i = (v_{i-1}, v_i)$$

\mathbf{v}_i denotes the position vector of v_i , and the edge vectors are defined as $\mathbf{e}_i = \mathbf{v}_i - \mathbf{v}_{i-1}$.

Definition 2 (Rotationally Symmetric Polyhedron). The rotationally symmetric polyhedra discussed in this paper (Fig. 2(b)) are described by n rotations by angle $2\pi/n$ of a profile about the z -axis.

Let (x_i, y_i, z_i) denote the coordinates of \mathbf{v}_i , where $\mathbf{v}_i \in \mathbb{R}^3$ denotes the position vector of $v_i \in V$. Since the profile cannot cross the z -axis, its vertices are restricted to the non-negative x half-plane: $x_i \geq 0, y_i = 0 \forall i \in \{0, 1, \dots, m\}$. The first vertex $\mathbf{v}_0 = (0, 0, z_0)$ must lie on the z -axis and the last vertex $\mathbf{v}_m = (x_m, 0, 0)$ must lie on the x -axis.

Each vertex v_i of the profile is also a vertex of a regular n -gon P_i centered at and normal to the z -axis, homothetic to the other regular polygons. Thus, if $\mathcal{H}(S)$ denotes the convex hull of the point set $S \subseteq \mathbb{R}^3$,

$$P_i = \mathcal{H}\left(\{(x_i \cos(k2\pi/n), x_i \sin(k2\pi/n), z_i) : k \in \mathbb{Z}\}\right) \quad (2.2)$$

$$\forall i \in \{0, 1, \dots, m\}.$$

Lemma 1. Each edge of P_i has length $2x_i \sin(\pi/n)$.

Definition 3 (Frustum, plural *Frusta*). Each frustum F_i (Fig. 3), where $i \in \{1, 2, \dots, m\}$, is bounded by two such polygons, P_{i-1} and P_i (the frustum's roof and base respectively); more precisely, $F_i = \mathcal{H}(P_{i-1} \cup P_i)$. Frustum F_i is *positive* if it rises up from the paper (i.e. $z_i \leq z_{i-1}$) and *negative* if it sinks into the paper (i.e. $z_i > z_{i-1}$). The side faces of the frustum will be referred to as *walls*.

The entire rotationally symmetric polyhedron is the union of every frustum, or $(\bigcup_{i=1}^m F_i)$.

Definition 4 (Crease Pattern and Crease). A crease pattern is a set of creases, where a crease $c = (\ell, \omega)$ consists of a line segment or ray

ℓ in the xy plane and its *polarity* as either a mountain ($\omega = 1$, black line) or valley fold ($\omega = -1$, red line) (e.g. see Fig. 4). Let \mathcal{C} denote the set of all creases, and let the crease pattern of F_i be denoted as C_i .

Definition 5 (Pleat). A pleat is a pair of a parallel mountain crease ($\ell_m, 1$) and valley crease ($\ell_v, -1$) which, when folded together, keep the paper flat. Formally, it is defined as a tuple $(\ell_m, \ell_v, \mathbf{n}, \omega)$ where the vector \mathbf{n} is normal to both creases, pointing from ℓ_m to ℓ_v , and $\|\mathbf{n}\|$ is twice the distance between ℓ_m and ℓ_v .

$\omega = 1$ (respectively -1) denotes that ℓ_m (respectively ℓ_v) is held stationary when the pleat is being folded or unfolded. \mathcal{P} denotes the set of all pleats:

$$\mathcal{P} = \{(\ell, \{\mathbf{u} + \mathbf{n}/2 : \mathbf{u} \in \ell\}, \mathbf{n}, \omega) : (\ell, \omega) \in \mathcal{C}, \mathbf{n} \in \mathbb{R}^2, \ell \perp \mathbf{n}\}. \quad (2.3)$$

The normal vector \mathbf{n} is stored for efficiency reasons, since it is required multiple times in later calculations. Recalculating it from the creases repeatedly would be unfavorable. Its length was chosen because most calculations require a normal vector scaled to that length.

3. Frustum crease patterns

A positive frustum can be folded as follows: each frustum is first unfolded into a net by removing the base, cutting along every edge connected to the base and flattening it. Small crease patterns (*gadgets*) are then inserted into the gap between adjacent walls to hide that paper inside the frustum upon folding, allowing the walls to close up. This avoids the problem regarding external flaps. The steps involved in folding a positive frustum are listed below and illustrated in Fig. 5.

- i. An extruded pentagonal frustum ($n = 5$).
- ii. The walls are unfolded and the frustum is flattened.
- iii. The following steps draw creases between every pair of adjacent walls to form a gadget.
- iv. The bounding segments are extended into rays. Perpendiculars are extended until they meet.
- v. Their perpendicular bisectors (red rays) are drawn.
- vi. Reflect the walls' bottom edges along the black rays.
- vii. The intersection points are connected and irrelevant parts of lines are erased.
- viii. Steps iv–vii are repeated between every pair of adjacent walls.

Fig. 6 demonstrates the folding of a positive frustum from its crease pattern. Fig. 6(b)–(d) demonstrate a gadget being folded to hide the paper inside the frustum. Fig. 6(e)–(f) show the other gadgets being folded in the same way.

Another procedure has also been derived in this paper to draw the crease pattern of a negative frustum. The steps involved are listed below and illustrated in Fig. 7.

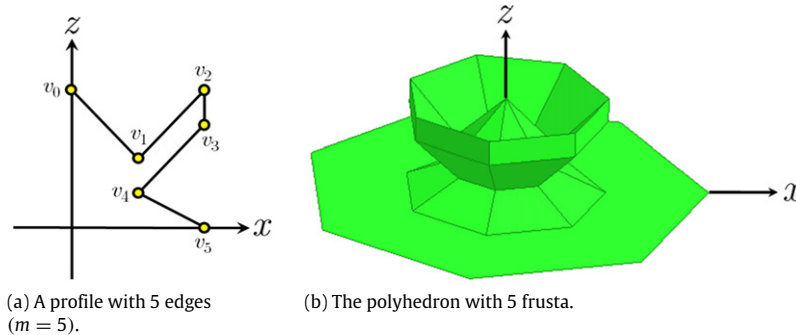


Fig. 2. A profile and its rotationally symmetric polyhedron ($n = 7$).

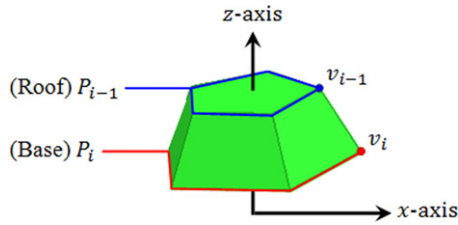


Fig. 3. Positive frustum F_i . The top pentagon is P_{i-1} and the bottom pentagon (partially obscured) is P_i .

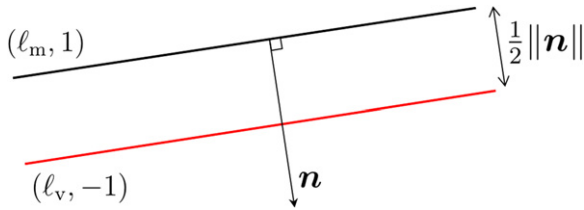


Fig. 4. A pleat $(\ell_m, \ell_v, n, \omega)$. Its creases are $(\ell_m, 1)$ and $(\ell_v, -1)$.

- i. A negative pentagonal frustum ($n = 5$).
- ii. The walls are unfolded and the frustum is flattened.
- iii. The following steps draw creases between every pair of adjacent walls to form a gadget.

- iv. Perpendiculars are extended until they meet.
- v. Their perpendicular bisectors (red rays) are drawn.
- vi. Parallel black rays are extended from the intersection between the dotted lines.
- vii. Horizontal lines are drawn in the line of symmetry.
- viii. The bottom edge of one of the walls is extended to the red ray.
- ix. The extension is reflected as shown, switching polarity each time.
- x. A red segment is drawn as shown such that the marked lengths are equal.
- xi. The segment is reflected across the red ray.
- xii. Steps iv–xi are repeated between every pair of adjacent walls.

Fig. 8 demonstrates the folding of a negative frustum from its crease pattern. **Fig. 8(b)–(c)** demonstrate a gadget being folded to hide the paper beneath the external surface. **Fig. 8(d)–(e)** show the other gadgets being folded in the same way.

Theorem 1. Given that α_i is the internal angle (touching the roof P_{i-1}) of any wall (Figs. 5–7),

$$\alpha_i = \cos^{-1}((x_{i-1} - x_i) \sin(\pi/n) / \|\mathbf{e}_i\|). \quad (3.1)$$

Proof. α_i is bounded by an edge \mathbf{w} of P_{i-1} and an edge \mathbf{e}_i of the profile that connects the roof and the base. By **Lemma 1**, we have

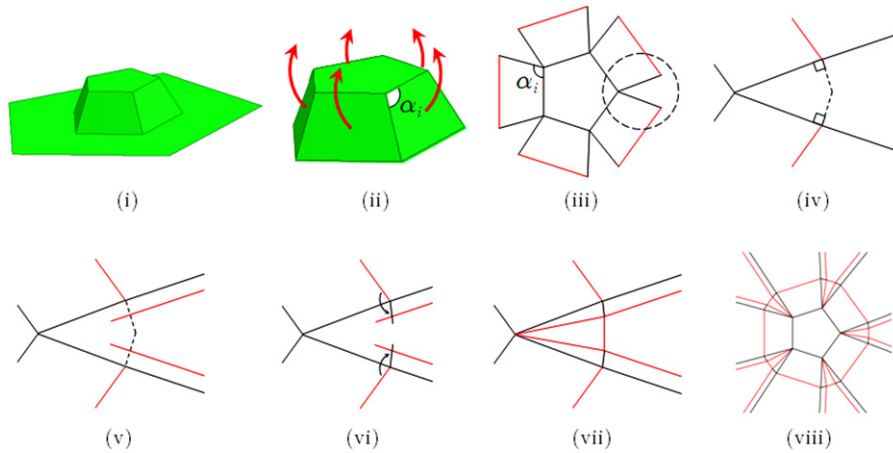


Fig. 5. Procedure to draw a positive frustum's crease pattern.

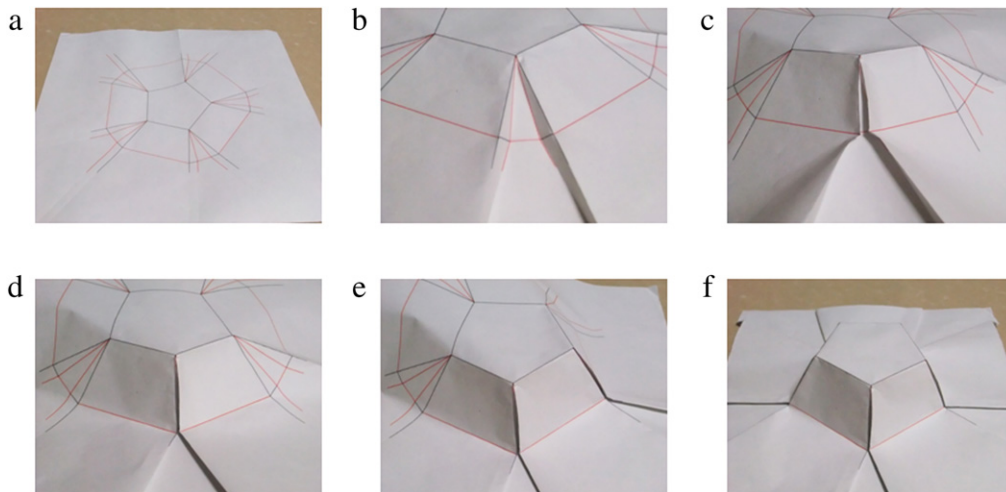


Fig. 6. Folding a positive frustum from its crease pattern.

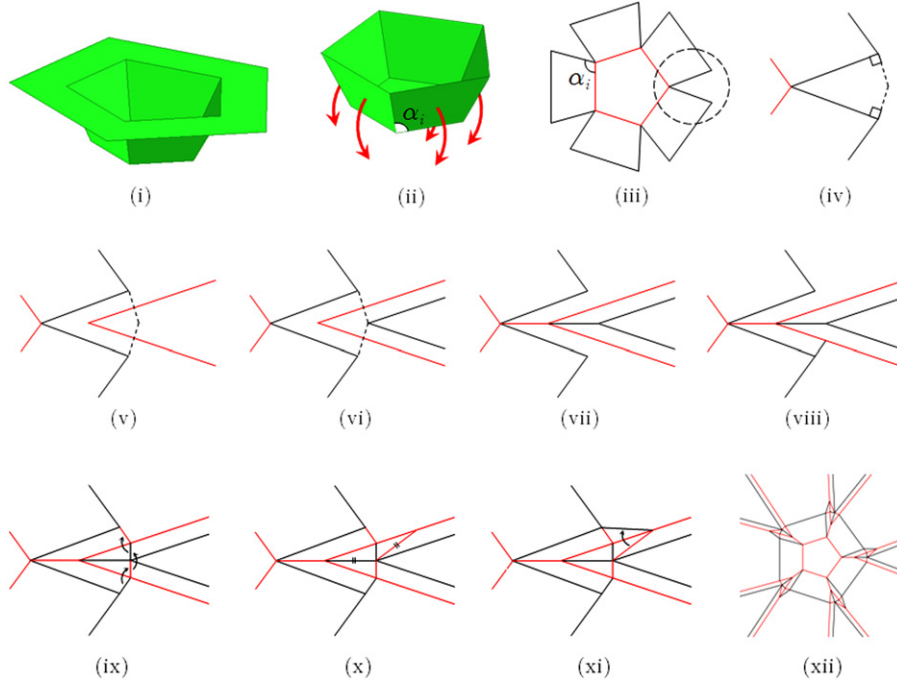


Fig. 7. Procedure to draw a negative frustum's crease pattern.

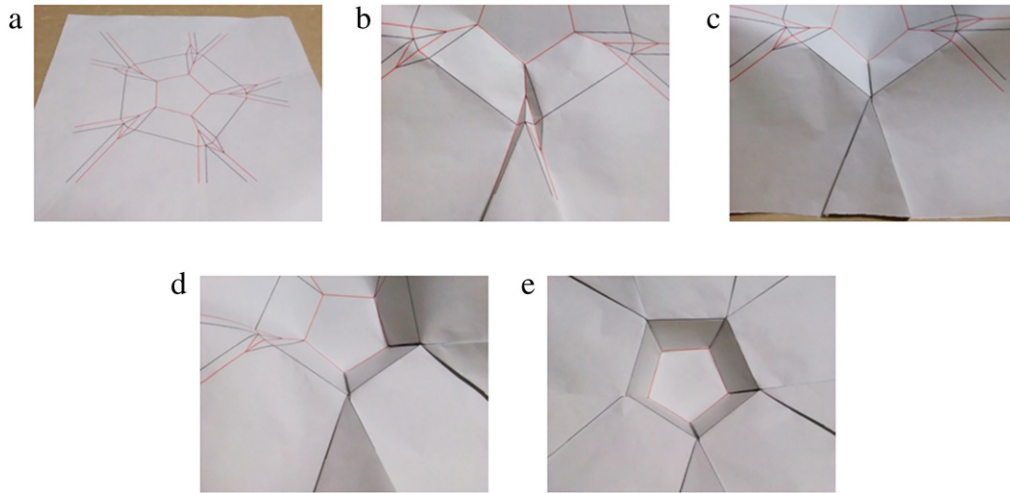


Fig. 8. Folding a negative frustum from its crease pattern.

$\|\mathbf{w}\| = 2x_{i-1} \sin(\pi/n)$. If $\text{Rot}(\theta)$ denotes a rotation in the xy plane by counterclockwise angle θ ,

$$\begin{aligned}
 \mathbf{w} &= \text{Rot}(2\pi/n)\mathbf{v}_{i-1} - \mathbf{v}_{i-1} \\
 \Rightarrow \cos \alpha_i &= \hat{\mathbf{w}} \cdot \hat{\mathbf{e}}_i \\
 &= \frac{(\text{Rot}(2\pi/n)\mathbf{v}_{i-1} - \mathbf{v}_{i-1}) \cdot (\mathbf{v}_i - \mathbf{v}_{i-1})}{2x_{i-1} \sin(\pi/n)\|\mathbf{e}_i\|} \\
 &= \frac{\begin{bmatrix} x_{i-1}(\cos(2\pi/n) - 1) \\ x_{i-1} \sin(2\pi/n) \\ 0 \end{bmatrix} \cdot \begin{bmatrix} x_i - x_{i-1} \\ 0 \\ z_i - z_{i-1} \end{bmatrix}}{2x_{i-1} \sin(\pi/n)\|\mathbf{e}_i\|} \\
 &= \frac{x_{i-1}(x_i - x_{i-1})(-2 \sin^2(\pi/n))}{2x_{i-1} \sin(\pi/n)\|\mathbf{e}_i\|} \\
 &= (x_{i-1} - x_i) \sin(\pi/n) / \|\mathbf{e}_i\| \\
 \Rightarrow \alpha_i &= \cos^{-1}((x_{i-1} - x_i) \sin(\pi/n) / \|\mathbf{e}_i\|). \quad \square
 \end{aligned} \tag{3.2}$$

Corollary 1. Any non-vertical edge e_i of the profile will have a gradient $\gamma \in \mathbb{R}$ in the xz plane, with

$$\alpha_i = \cos^{-1} \left(-\frac{\sin(\pi/n)}{\sqrt{1 + \gamma^2}} \right). \tag{3.3}$$

This results from a simplification of the expression for α_i from Theorem 1:

$$\|\mathbf{e}_i\| = \sqrt{(x_i - x_{i-1})^2 + (z_i - z_{i-1})^2} \tag{3.4}$$

$$\begin{aligned}
 \Rightarrow \alpha_i &= \cos^{-1} \left(\frac{(x_{i-1} - x_i) \sin(\pi/n)}{\sqrt{(x_i - x_{i-1})^2 + (z_i - z_{i-1})^2}} \right) \\
 &= \cos^{-1} \left(-\frac{\sin(\pi/n)}{\sqrt{1 + \gamma^2}} \right).
 \end{aligned} \tag{3.5}$$

Note that even if the roof degenerates to a point (i.e. $x_i = 0$), the positive and negative frustum algorithms can still be applied by

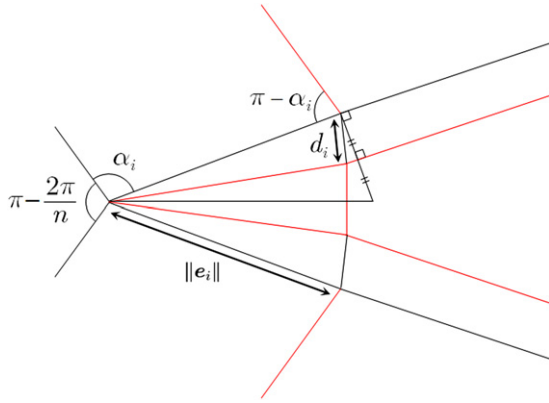


Fig. 9. Predicting the intersection of hidden paper.

treating P_{i-1} as an infinitesimal n -gon. This treatment also allows α_i to be defined.

In the case of the positive frustum algorithm, the paper hidden inside the frustum by each gadget may intersect if each gadget is too large, and the frustum cannot be folded properly. This can be predicted by finding a certain length d_i of the gadget as defined in Fig. 9.

It can be shown that

$$\tan\left(\frac{2\pi - (\pi - \frac{2\pi}{n}) - 2\alpha_i}{2}\right) = \frac{2d_i \cos(\pi/2 - (\pi - \alpha_i))}{\|e_i\|} \quad (3.6)$$

$$\Rightarrow \tan\left(\frac{\pi}{2} + \frac{\pi}{n} - \alpha_i\right) = \frac{2d_i \cos(\alpha_i - \pi/2)}{\|e_i\|} \quad (3.7)$$

$$\Rightarrow d_i = \frac{\|e_i\|}{2 \sin \alpha_i \tan(\alpha_i - \pi/n)}. \quad (3.8)$$

Intersection of hidden paper can be solved by splitting F_i into q frusta using $(q-1)$ cutting planes evenly spaced between P_{i-1} and P_i , scaling d_i by factor $1/q$.

Lemma 2. If q_i denotes the smallest number of frusta that F_i must be cut into in order to solve the intersection of hidden paper, then

$$q_i = \left\lceil \frac{d_i / \sin(\pi/n) - x_i}{x_{i-1}} \right\rceil + 1. \quad (3.9)$$

Proof. Since each edge of the base P_i has two gadgets taking up space along it, each gadget can take up at most half the length of that edge, which by Lemma 1 is $x_i \sin(\pi/n)$. Hence, to solve intersection, d_i must be scaled by factor $1/q$ for some integer q such that

$$\frac{d_i}{q} < \left(x_{i-1} + \frac{x_i - x_{i-1}}{q}\right) \sin(\pi/n) \\ \Rightarrow q > \frac{d_i / \sin(\pi/n) - x_i}{x_{i-1}} + 1 \quad (3.10)$$

$$\therefore q_i = \left\lceil \frac{d_i / \sin(\pi/n) - x_i}{x_{i-1}} \right\rceil + 1. \quad (3.11)$$

The creases in C_i can be categorized as the central body (which is extruded after folding) and the pleats radiating out from it (Fig. 10). The central body has n -fold rotational symmetry, which can be exploited as discussed later in the paper. The pleats from C_i (either a positive or negative frustum's crease pattern) are indexed as shown in Fig. 10. The pleat radiating from v_i into the positive y half-plane is $p_{(i,1)}^+$; the other pleats are indexed as follows:

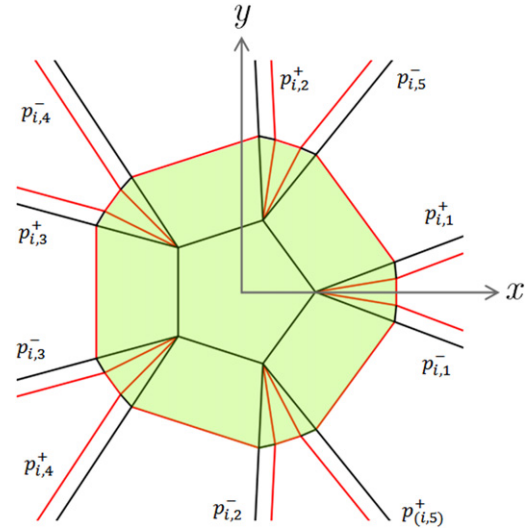


Fig. 10. The shaded region is the central body of frustum F_i ($n = 5$). The pleats of F_i radiate out from it.

$$(\text{positive pleats}) p_{i,j}^+ = \text{Rot}((j-1)2\pi/n)p_{i,1}^+ \quad \forall j \in \mathbb{Z} \quad (3.12)$$

$$(\text{negative pleats}) p_{i,k}^- = \text{Ref}(x\text{-axis})p_{i,k}^+ \quad \forall k \in \mathbb{Z} \quad (3.13)$$

where $\text{Ref}(\ell)$ denotes reflection about the line ℓ .

Note that according to Eq. (3.12), $p_{i,a}^+$ and $p_{i,b}^+$ refer to the same pleat if and only if $a \equiv b \pmod{n}$; this allows for $p_{i,j-k}^+$ and $p_{i,j+k}^+$ ($k \in \mathbb{N}$) to correctly refer to preceding and succeeding pleats respectively. To illustrate, this notation allows $p_{i,2}^+ = p_{i,(n-1)+3}^+$ to refer to the third pleat anticlockwise from $p_{i,n-1}^+$ (see Fig. 10). The indices of $p_{i,j \pm k}^-$ and all other cyclically ordered objects in this report will be defined similarly.

4. Crease pattern composition algorithm

Algorithm 1 outlines the procedure drawing a rotationally symmetric polyhedron's crease pattern:

Algorithm 1

Rotationally Symmetric Polyhedron Crease Pattern Algorithm

- 1: **for** $i = 1$ **to** m **do**
- 2: Ignore any previously folded frusta and treat the paper as if it were new.
- 3: Draw C_i and fold it to extrude F_i .
- 4: **end for**
- 5: **for** $j = m$ **to** 1 **do**
- 6: Unfold F_j .
- 7: **end for**

The positive and negative algorithms allow frusta to be folded from flat sheets. In Algorithm 1, each frustum is folded using the flat paper surrounding the extruded region of the previous frustum without interfering with the extruded region, so this ensures that the target rotationally symmetric polyhedron is folded after Line 4 of Algorithm 1.

The process of folding one crease pattern on another, and then unfolding both, causes the *composition* of the two crease patterns. The crease patterns from both the positive and negative frustum algorithms have pleats that extend outward from the frustum body. Pleats complicate the composition of crease patterns, since they break and shift the surface of the paper. For brevity, the composition of two frustum crease patterns will be denoted by the binary operator $\odot : 2^e \times 2^e \rightarrow 2^e$. \odot is considered as

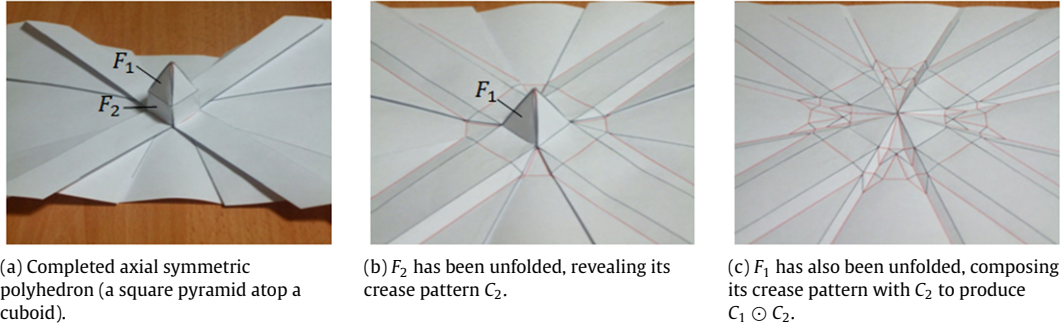
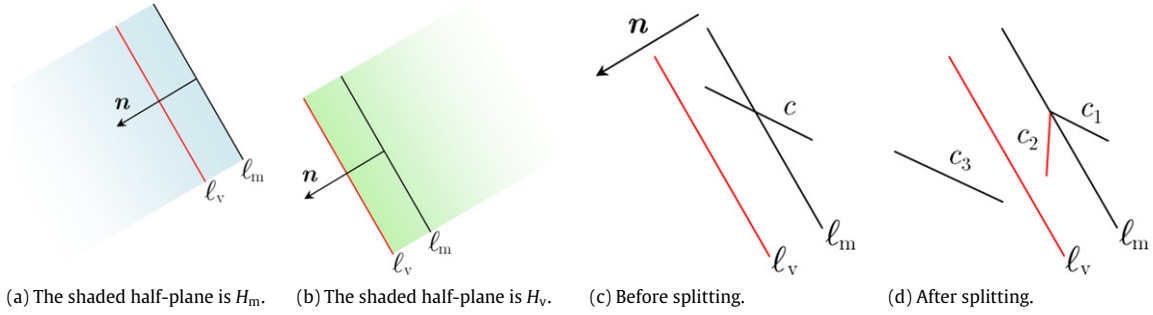


Fig. 11. Unfolding frusta.

Fig. 12. A pleat $(\ell_m, \ell_v, \mathbf{n}, 1)$ splitting a crease c .

right-associative to assign the order of evaluation without using parentheses. When the frusta are unfolded in Algorithm 1 (Line 6), the bottommost frustum F_m is unfolded first, revealing its crease pattern C_m on the paper. The next frustum F_{m-1} is then unfolded, and in the process C_m is broken up by the unfolding of the pleats from C_{m-1} . The pieces of C_m are translated to make way for C_{m-1} . The crease pattern on the paper at this stage is $C_{m-1} \oplus C_m$, as illustrated in Fig. 11.

This process is repeated until all frusta are unfolded; it is evident that the output of Algorithm 1 is $C_1 \oplus C_2 \oplus \dots \oplus C_m$.

4.1. Splitting of creases by pleats

A large part of the crease pattern arises from the splitting of the crease patterns by pleats. Suppose a pleat was folded, a new crease c was folded on it and everything was then unfolded; three new creases (c_1 , c_2 and c_3) would show on the paper, induced by c (Fig. 12(c)–(d)).

c_1 , c_2 and c_3 will be formally described for computation in the algorithm. In order to specify parts of a line in relation to pleats (which have the form $(\ell_m, \ell_v, \mathbf{n}, \pm 1)$), the half-plane H_m is defined as the half-plane bounded by ℓ_m and containing ℓ_v (Fig. 12(a)), and H_v is defined as the half-plane bounded by ℓ_v and containing ℓ_m (Fig. 12(b)).

$$H_m = \{\mathbf{x} + \lambda \mathbf{n} : \mathbf{x} \in \ell_m, \lambda \in \mathbb{R}^+\} \quad \text{and} \quad (4.1)$$

$$H_v = \{\mathbf{y} + \mu \mathbf{n} : \mathbf{y} \in \ell_v, \mu \in \mathbb{R}^-\}.$$

Lemma 3. From Definition 5, given a pleat $(\ell_m, \ell_v, \mathbf{n}, \omega)$,

$$\text{Ref}(\ell_v) \text{Ref}(\ell_m) = \text{Trans}(\mathbf{n}) \quad (4.2)$$

where $\text{Trans}(\mathbf{v})$ denotes translation by vector \mathbf{v} .

Lemma 4.

$$\text{Ref}(\ell_m) \text{Ref}(\ell_v) = \text{Trans}(-\mathbf{n}). \quad (4.3)$$

Theorem 2. A pleat $(\ell_m, \ell_v, \mathbf{n}, 1)$ will split a crease $c = (\ell, \omega)$ into the creases c_1 , c_2 and c_3 (see Fig. 12 (c)–(d)) where

$$c_1 = (\ell \cap H'_m, \omega) \quad (4.4)$$

$$c_2 = (\text{Ref}(\ell_m)\ell \cap H_m \cap H_v, -\omega) \quad (4.5)$$

$$c_3 = (\text{Trans}(\mathbf{n})\ell \cap H'_v, \omega). \quad (4.6)$$

Proof. Let S' denote the complement of the point set S in the xy plane, or $\mathbb{R}^2 \setminus S$. The paper can be partitioned by the pleat into the regions H'_m , $H_m \cap H_v$ and H'_v ; when the pleat is folded (in this case keeping ℓ_m stationary), these three regions are mapped as follows:

$$H'_m \xrightarrow{\text{fold}} H'_m \quad (4.7)$$

$$H_m \cap H_v \xrightarrow{\text{fold}} \text{Ref}(\ell_m)(H_m \cap H_v) \quad (4.8)$$

$$H'_v \xrightarrow{\text{fold}} \text{Ref}(\ell_m) \text{Ref}(\ell_v)H'_v = \text{Trans}(-\mathbf{n})H'_v. \quad (4.9)$$

When a crease (ℓ, ω) is folded on top of the folded pleat, the three creases it induces on each of the regions are $(\ell \cap H'_m, \omega)$, $(\ell \cap \text{Ref}(\ell_m)(H_m \cap H_v), \omega)$ and $(\ell \cap \text{Trans}(-\mathbf{n})H'_v, \omega)$. Unfolding the pleat then maps each of these induced creases with the respective inverse transformations which map the regions back into their original positions. Let id denote the identity transformation.

$$(\ell \cap H'_m, \omega) \xrightarrow[\text{unfold}]{\text{id}} (\ell \cap H'_m, \omega) \quad (4.10)$$

$$(\ell \cap \text{Ref}(\ell_m)(H_m \cap H_v), \omega) \quad (4.11)$$

$$\xrightarrow[\text{unfold}]{\text{Ref}(\ell_m)} (\text{Ref}(\ell_m)\ell \cap H_m \cap H_v, -\omega) \quad (4.11)$$

$$(\ell \cap \text{Trans}(-\mathbf{n})H'_v, \omega) \xrightarrow[\text{unfold}]{\text{Trans}(\mathbf{n})} (\text{Trans}(\mathbf{n})\ell \cap H'_v, \omega). \quad (4.12)$$

Eq. (4.11) shows the mapped induced crease having its polarity negated; this is because the paper was flipped over when unfolded, turning a mountain fold to a valley fold and vice versa. \square

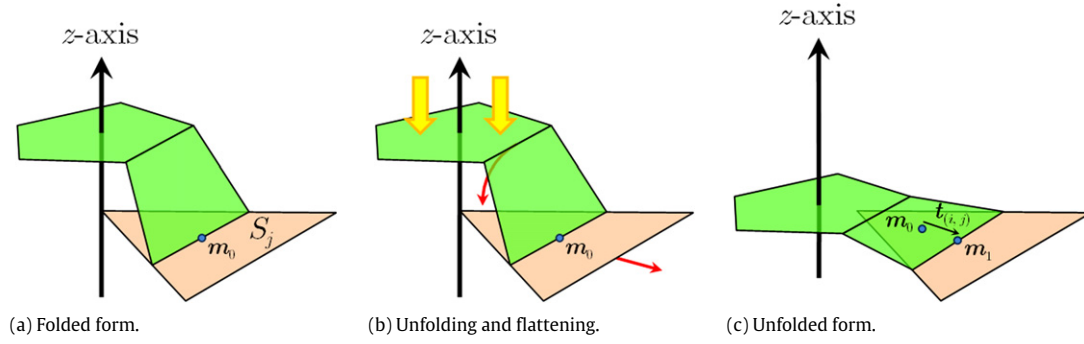


Fig. 13. Translating sector S_j radially away from the origin.

Remark.

If the function $\text{Split} : \mathcal{P} \times \mathcal{C} \rightarrow 2^{\mathcal{C}}$ is defined to produce the set of induced creases when a new crease (the second argument) is split by a pleat (the first argument), then

$$\begin{aligned} &\text{Split}((\ell_m, \ell_v, \mathbf{n}, 1), (l, \omega)) \\ &= \{(\ell \cap H'_m, \omega), (\text{Ref}(\ell_m)\ell \cap H_m \cap H_v, -\omega), \\ &\quad (\text{Trans}(\mathbf{n})\ell \cap H'_v, \omega)\}. \end{aligned} \quad (4.13)$$

Theorem 2 describes the implementation of crease splitting by pleats whose mountain crease is held stationary. If the valley crease was held stationary instead, the result is the same as if the mountain crease was stationary but the crease being split was translated by $-\mathbf{n}$, hence we have **Lemma 5**.

Lemma 5.

$$\begin{aligned} &\text{Split}((\ell_m, \ell_v, \mathbf{n}, -1), (l, \omega)) \\ &= \text{Split}((\ell_m, \ell_v, \mathbf{n}, 1), (\{\mathbf{u} - \mathbf{n} : \mathbf{u} \in \ell\}, \omega)). \end{aligned} \quad (4.14)$$

For efficient computation of the induced creases, the transformations applied should be expanded and combined. This leads us to **Proposition 1**.

Proposition 1.

$$\begin{aligned} &\text{Split}((\ell_m, \ell_v, \mathbf{n}, -1), (l, \omega)) = \{(\text{Trans}(-\mathbf{n})\ell \cap H'_m, \omega), \\ &\quad (\text{Ref}(\ell_v)\ell \cap H_m \cap H_v, -\omega), (\ell \cap H'_v, \omega)\}. \end{aligned} \quad (4.15)$$

Proof. By **Lemma 5**,

$$\begin{aligned} &\text{Split}((\ell_m, \ell_v, \mathbf{n}, -1), (l, \omega)) \\ &= \text{Split}((\ell_m, \ell_v, \mathbf{n}, 1), (\text{Trans}(-\mathbf{n})\ell, \omega)) \end{aligned} \quad (4.16)$$

$$= \{c_1, c_2, c_3\} \quad (4.17)$$

where

$$\begin{aligned} c_1 &= (\text{Trans}(-\mathbf{n})\ell \cap H'_m, \omega) \\ c_2 &= (\text{Ref}(\ell_m) \text{Trans}(-\mathbf{n})\ell \cap H_m \cap H_v, -\omega) \\ &= (\text{Ref}(\ell_m)(\text{Ref}(\ell_m) \text{Ref}(\ell_v))\ell \cap H_m \cap H_v, -\omega) \\ &= (\text{Ref}(\ell_v)\ell \cap H_m \cap H_v, -\omega) \\ c_3 &= (\text{Trans}(\mathbf{n}) \text{Trans}(-\mathbf{n})\ell \cap H'_v, \omega) \\ &= (\ell \cap H'_v, \omega). \quad \square \end{aligned}$$

4.2. Optimizing the algorithm: partitioning crease patterns into sectors

Since all pleats are rays, they only split creases that they intersect; thus the creases that a pleat can split should be restricted. This is done by partitioning the crease pattern into n sectors and only splitting the creases in certain sectors. The region of the xy plane belonging to each sector $\delta_1, \delta_2, \dots, \delta_n$ is defined as follows:

$$\begin{aligned} \delta_j &= \{(r \cos \theta, r \sin \theta) : r \in \mathbb{R}^+, \theta \in [(j-1)2\pi/n, j2\pi/n)\} \\ &\quad \forall j \in \mathbb{Z}. \end{aligned} \quad (4.18)$$

Any crease pattern C can be partitioned according to the sectors into an n -tuple of crease patterns (S_1, S_2, \dots, S_n) such that

$$S_j = \{(\ell \cap \delta_j, \omega) : (\ell, \omega) \in C\}. \quad (4.19)$$

$\delta_1, \delta_2, \dots, \delta_n$ are defined solely to partition crease patterns. The output of the partitioning, S_1, S_2, \dots, S_n , are parts of actual crease patterns that are operated on subsequently.

In order to optimize the efficiency of the algorithm, the repeated partitioning of the crease pattern into sectors (which has complexity $O(n|C|)$) should be avoided. Hence, each sector of the crease pattern is monitored separately (as an n -tuple of crease patterns) and never unioned together. This removes the need to partition at all, improving the complexity of that step from $O(n|C|)$ to $O(1)$. This isolation is possible because the operations performed on one sector do not create creases that belong in another.

The crease pattern composition is slightly modified as follows: a right-associative external binary operator $\odot : 2^{\mathcal{C}} \times (2^{\mathcal{C}})^n \rightarrow (2^{\mathcal{C}})^n$ is defined, modifying the current tuple of sectors (the second argument) based on the crease pattern in the first argument. The following work derives the steps in Algorithm 2, which computes one crease pattern composition, i.e. $C_i \odot (S_1, S_2, \dots, S_n)$.

Theorem 3. The unfolding of frustum F_i causes each sector S_j to be translated radially away from the origin via the transformation $T_{i,j} : 2^{\mathcal{C}} \rightarrow 2^{\mathcal{C}}$.

Proof. From Fig. 13, it can be seen that each sector S_j is translated away from the origin in the same direction (radially outwards) as the edge of the base P_i which lies inside the sector. By the symmetry in the frustum, the unit displacement vector $\hat{\mathbf{t}}_{i,j}$ bisects the angle subtended by S_j :

$$\hat{\mathbf{t}}_{i,j} = \text{Rot}((2j-1)\pi/n)\hat{\mathbf{i}} = \begin{bmatrix} \cos((2j-1)\pi/n) \\ \sin((2j-1)\pi/n) \end{bmatrix}. \quad (4.20)$$

The magnitude of the displacement $\|\mathbf{t}_{i,j}\|$ is the distance between the base edge and its position after unfolding the frustum measured in the perpendicular direction. The symmetry of the frustum makes the foot of perpendicular the same as the midpoint of the base edge; if the positions of the midpoint in the folded and unfolded frustum are \mathbf{m}_0 and \mathbf{m}_1 respectively, then their distances from the origin are as shown in Fig. 14, which depicts the cross-sections of Fig. 13(a) and (c) by a plane passing through the z -axis and \mathbf{m}_0 .

$$\|\mathbf{t}_{i,j}\| = \|\mathbf{m}_1\| - \|\mathbf{m}_0\| \quad (4.21)$$

$$= (x_{i-1} \cos(\pi/n) + \|\mathbf{e}_i\| \cos(\alpha_i - \pi/2)) - x_i \cos(\pi/n) \quad (4.22)$$

$$= \|\mathbf{e}_i\| \cos(\alpha_i - \pi/2) - (x_i - x_{i-1}) \cos(\pi/n). \quad (4.23)$$

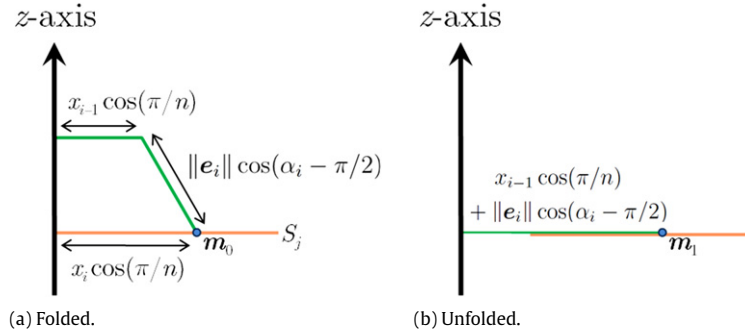
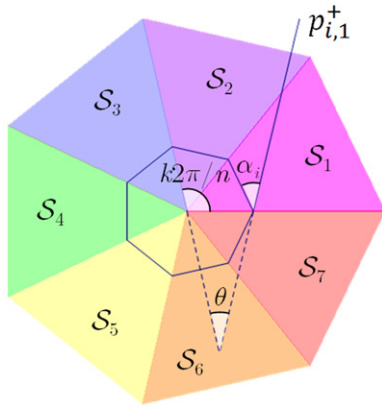


Fig. 14. Tracking the midpoint of the base edge.

Fig. 15. The sectors which a pleat can reach ($k = 2$).

Hence, if $\text{Trans}(\mathbf{v})$ denotes a translation by vector \mathbf{v} , then

$$T_{i,j}(S_j) = \text{Trans}(\mathbf{t}_{i,j})S_j \quad (4.24)$$

where

$$\mathbf{t}_{i,j} = \left(\|e_i\| \cos(\alpha_i - \pi/2) - (x_i - x_{i-1}) \cos(\pi/n) \right) \begin{bmatrix} \cos((2j-1)\pi/n) \\ \sin((2j-1)\pi/n) \end{bmatrix}. \quad \square \quad (4.25)$$

For each frustum F_i , the number s_i of sectors which a pleat can intersect is fixed. Fig. 15 illustrates an example where the heptagon ($n = 7$) in the center is the roof of F_i . A small value of α_i was chosen in Fig. 15 to illustrate an s_i value greater than 1. In that case, $s_i = 2$ because each pleat can only intersect two sectors.

Proposition 2.

$$s_i = \left\lceil \frac{n}{4} + \frac{1}{2} - \frac{n\alpha_i}{2\pi} \right\rceil. \quad (4.26)$$

Proof. The boundaries between sectors form an angle $k2\pi/n$ with the x -axis for some $k \in \mathbb{Z}$. θ being positive for some k implies that the pleat does not intersect that boundary, i.e. the pleat cannot intersect sectors beyond that boundary. The pleat intersects all sectors within the boundary only when k is at its minimum, hence s_i is the minimum k such that $\theta > 0$. Recall that α_i is the internal angle (touching F_i 's roof) of F_i 's walls; it can be shown that

$$\theta = \pi - \left(\pi - \frac{k2\pi}{n} \right) - \left(\pi - \alpha_i - \left(\frac{\pi}{2} - \frac{\pi}{n} \right) \right) > 0 \quad (4.27)$$

$$\implies k > \frac{n}{4} + \frac{1}{2} - \frac{n\alpha_i}{2\pi} \quad (4.28)$$

$$\implies s_i = \left\lceil \frac{n}{4} + \frac{1}{2} - \frac{n\alpha_i}{2\pi} \right\rceil. \quad \square \quad (4.29)$$

4.3. Intersections among pleats

When computing $C_i \odot (S_1, S_2, \dots, S_n)$, pleats from C_i not only intersect creases in the sectors, but they may also intersect each other. The positive pleats can never intersect with each other, and similarly for the negative pleats, but negative and positive pleats will intersect if and only if $\alpha_i < \pi/2$. Hence, when $\alpha_i < \pi/2$, the creases in the sectors are split using positive pleats first, and the creases of the positive pleats themselves are added to the sectors so the negative pleats can split them, after which the creases of the negative pleats are added. Otherwise, the creases of all pleats are added only after all splitting is done.

Other than pleats, C_i has a central body of creases; its rotational symmetry implies that the portions of the central body inside each sector are congruent to one another via rotation. Let the portion inside S_1 be Γ_i .

Lemma 6. The portion of the central body inside S_j is $\text{Rot}((j-1)2\pi/n)\Gamma_i$.

The pleats and the central body form a partition of C_i :

$$\bigcup_{j=1}^n \left(\text{Rot}((j-1)2\pi/n)\Gamma_i \cup \Pi_{i,j}^+ \cup \Pi_{i,j}^- \right) = C_i \quad (4.30)$$

where the creases of the positive (respectively negative) pleats of C_i inside S_j are denoted by $\Pi_{i,j}^+$ (respectively $\Pi_{i,j}^-$).

Algorithm 2 shown on the next page consists of the procedures as described above.

The crease pattern of the entire rotationally symmetric polyhedron, is $C_1 \odot C_2 \odot \dots \odot C_m \odot t_0$ where $t_0 \in \{\emptyset\}^n$ is the initial tuple of empty crease patterns.

To summarize, the crease pattern of a rotationally symmetric polyhedron is generated using Algorithm 1. The state of the crease pattern after unfolding each frustum (Line 6, Algorithm 1) is generated by Algorithm 2.

5. Implementation and results

A CAD program has been written to implement our proposed algorithm for folding rotationally symmetric polyhedra. Users of the program draw a profile with m segments (precise coordinates of points may be specified) and select the number of rotations, n , via the user interface while verifying the specified solid which is displayed for the user's convenience. The crease pattern is automatically generated by the program and may be printed out to be folded. Examples 1–6 are rotationally symmetric polyhedra specially chosen from among convex and concave polyhedra for their varied structures to test the correctness and capabilities of the algorithm. Their crease patterns have been generated using the program and folded into the desired solids (Figs. 16–21).

Examples 1–3 are convex polyhedra. Example 1 (Fig. 16) has a regular polygon (hexagon) as its top face, which occurs when $z_0 = z_1$.

Algorithm 2 Computing $C_i \odot (S_1, S_2, \dots, S_n)$

```

1: for  $j = 1$  to  $n$  do
2:    $S_j \leftarrow T_{i,j}(S_j)$  (Theorem 3)
3: end for
4: for  $k = 1$  to  $n$  do
5:   for  $a = 1$  to  $s_i$  do
6:      $S_k \leftarrow \bigcup \{Split(p_{i,k-a+1}^+, c) : c \in S_k\}$  (Theorem 2 and Proposition 1)
7:   end for
8: end for
9: for  $l = 1$  to  $n$  do
10:  if  $\alpha_i < \pi/2$  then
11:     $S_l \leftarrow \Pi_{i,l}^+ \cup S_l$  (Section 4.3)
12:    for  $b = 1$  to  $s_i$  do
13:       $S_l \leftarrow \bigcup \{Split(p_{i,n+2-l-b}^-, c) : c \in S_l\}$  (Theorem 2 and Proposition 1)
14:    end for
15:  else
16:    for  $b = 1$  to  $s_i$  do
17:       $S_l \leftarrow \bigcup \{Split(p_{i,n+2-l-b}^-, c) : c \in S_l\}$  (Theorem 2 and Proposition 1)
18:    end for
19:     $S_l \leftarrow \Pi_{i,l}^+ \cup S_l$  (Section 4.3)
20:  end if
21: end for
22: for  $r = 1$  to  $n$  do
23:    $S_r \leftarrow Rot((r-1)2\pi/n) \Gamma_i \cup \Pi_{i,r}^- \cup S_r$  (Lemma 6)
24: end for

```

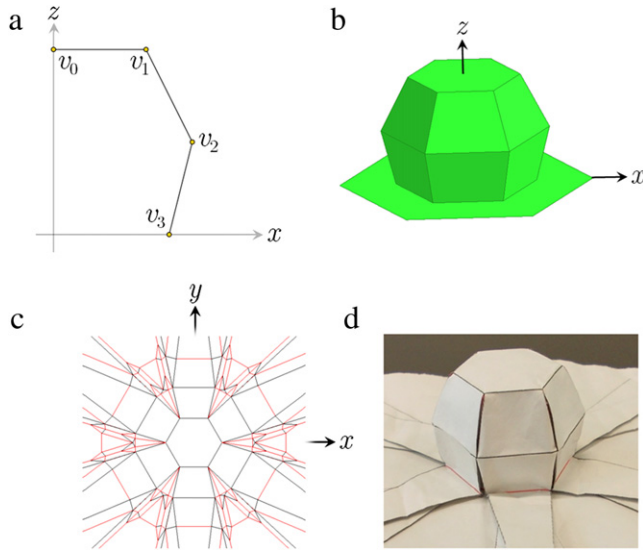


Fig. 16. Example 1 (a) The profile ($m = 3$). (b) The solid ($n = 6$). (c) The crease pattern generated by the program. (d) The folded product.

Example 2 (Fig. 17) has a pointed top, which occurs when $z_0 \neq z_1$.

Example 3 (Fig. 18) has an equilateral triangle as its top face since $z_0 = z_1$. Example 3 illustrates how the folded polyhedra can approximate curved surfaces using a profile that approximates a curve with many segments.

Examples 4–6 are concave polyhedra. Example 4 (Fig. 19) illustrates how concave solids with “wells” (sunken regions) in them can be folded using the algorithm.

Example 5 (Fig. 20) shows a concave solid with multiple wells. The rotational symmetry naturally causes the wells to form concentric “trenches”.

Example 6 (Fig. 21) is a more complex concave polyhedron that is composed of only positive frusta. Example 6 approximates

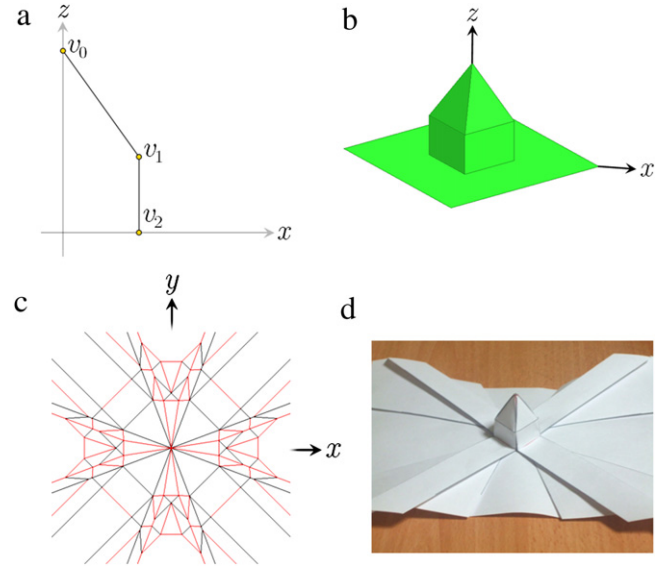


Fig. 17. Example 2 (a) The profile ($m = 2$). (b) The solid ($n = 4$). (c) The crease pattern generated by the program. (d) The folded product.

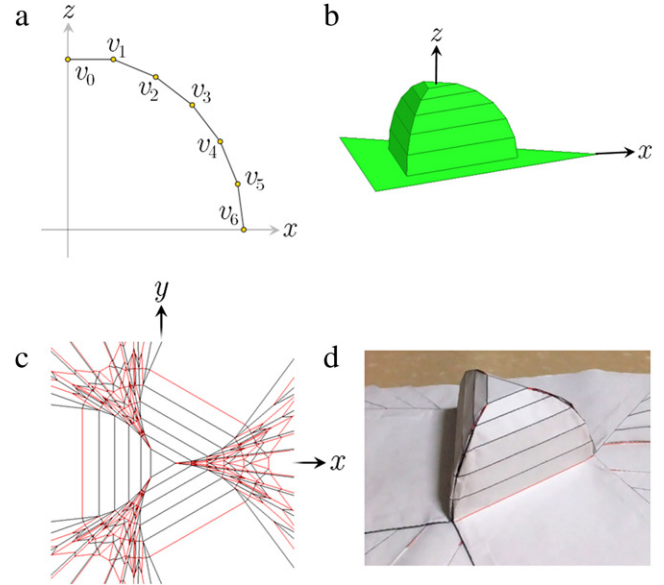


Fig. 18. Example 3 (a) The profile ($m = 6$). (b) The solid ($n = 3$). (c) The crease pattern generated by the program. (d) The folded product.

a concave curved surface since its profile has many segments to approximate a curve.




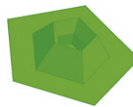


The parameters and data concerning the specification of the test examples and the generation of their crease patterns are summarized in Table 1.

Finally, we illustrate in Fig. 22 the folding process for Example 3 from its crease pattern. Here, the crease pattern of a rotational symmetric polyhedron is folded by consecutively folding individual frusta in the order F_1, F_2, \dots, F_m .

6. Folding with bicolor paper

If a piece of bicolor paper was used to fold the rotationally symmetric polyhedron, then the task of finding the distribution of the two colors on the external surface of the folded model arises. This question may have relevance in the practical application of the algorithm; for instance, the flat material used to fold a rotationally symmetric polyhedron may have different properties on each side.

Table 1

						
No. of segments in the profile, m	3	2	6	4	4	4
Order of rotational symmetry, n	6	4	3	5	4	6
No. of positive frusta	2	2	5	1	2	4
No. of negative frusta	0	0	0	2	2	0
Crease pattern computation time (s)	0.24	0.091	1.5	0.32	0.47	0.81
No. of creases	318	160	876	420	328	526
Example no.	1	2	3	4	5	6

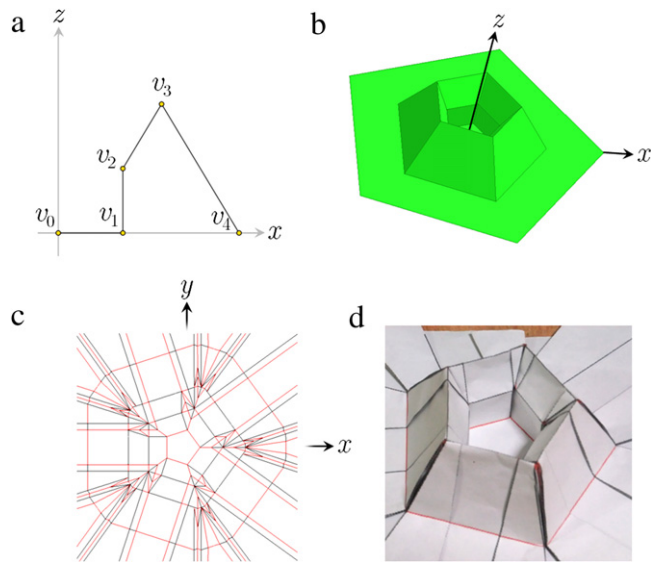


Fig. 19. Example 4 (a) The profile ($m = 4$). (b) The solid ($n = 5$). (c) The crease pattern generated by the program. (d) The folded product.

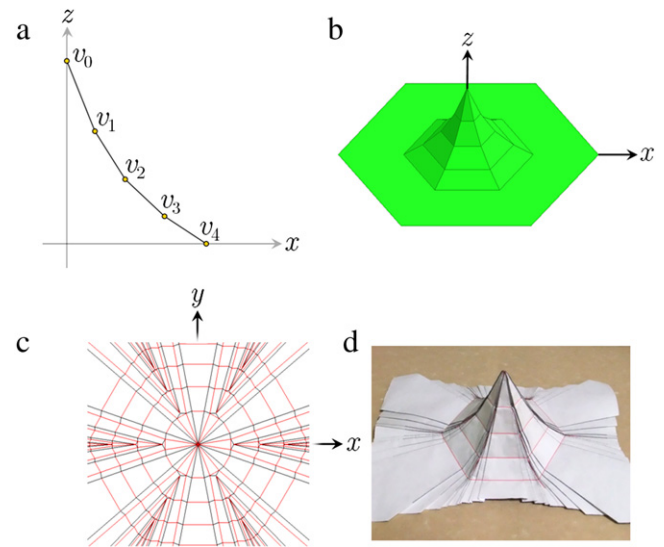


Fig. 21. Example 6 (a) The profile ($m = 4$). (b) The solid ($n = 6$). (c) The crease pattern generated by the program. (d) The folded product.

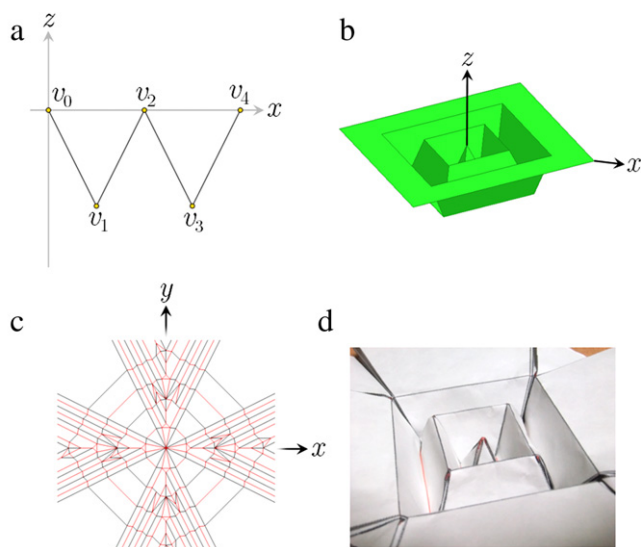


Fig. 20. Example 5 (a) The profile ($m = 4$). (b) The solid ($n = 4$). (c) The crease pattern generated by the program. (d) The folded product.

Thus, it may be useful to ensure that the external and internal surfaces of the folded model have uniform properties (and hence, color). If the external surface of the folded model has only one color and the internal surface only has another, then let us call the piece of bicolor paper used to fold the model a *color divider* of it. We have the following results.

Lemma 7. The piece of bicolor paper that spans the xy plane is a color divider of right frusta.

Proof. Without loss of generality, let the paper be colored white on top and black below. The external surface of the extruded solid must be white since it is formed by the walls of the frustum for both positive (Fig. 5(ii)–(iii)) and negative (Fig. 7(ii)–(iii)) frusta, which have white nets. The only parts of the paper surrounding the extruded polyhedron that are flipped over to show the black color are the regions between ℓ_m and ℓ_v in every pleat. Referring to Eq. (4.8), for a pleat with ℓ_m kept stationary, the region $(H_m \cap H_v)$ is flipped over (hence the region is reflected about ℓ_m) after folding, but the black area is covered by the white region H'_m . A similar argument applies for pleats with ℓ_v held stationary and hence also for every pleat in the surrounding paper. This means that the surrounding paper also has a white external surface. Likewise for the pleats as viewed from the internal surface, thereby proving that the internal surface is black. \square

Proposition 3. The piece of bicolor paper that spans the xy plane is a color divider of rotationally symmetric polyhedra.

Proof. By Lemma 7, after folding a right frustum, the surrounding paper serves as a new piece of paper colored white on top and black below. Since that surrounding paper is used to fold subsequent frusta, it can be shown inductively that the rotationally symmetric polyhedron (the result of composing frusta) also has a white external surface and black internal surface. \square

However, if the paper used is bounded, then at the intersections between pleats and the boundary of the paper, black areas may

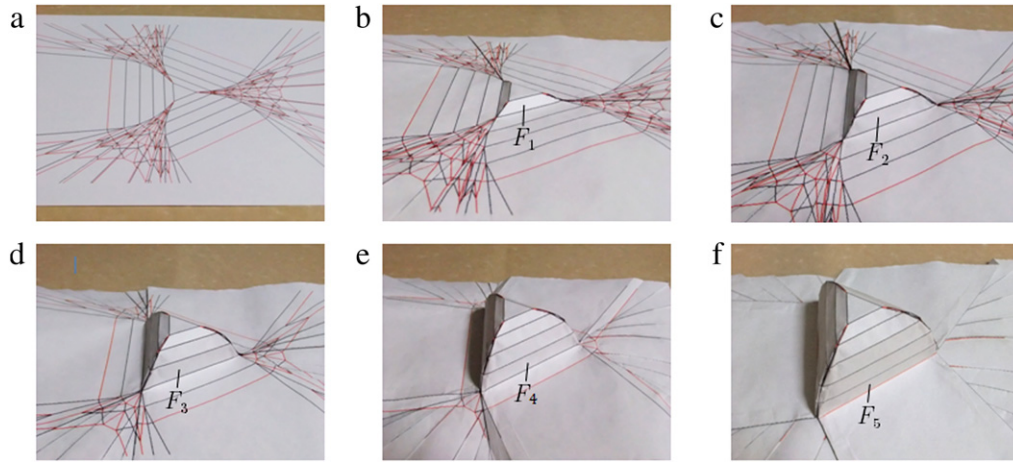


Fig. 22. Folding a crease pattern into the final polyhedron ($m = 5$).

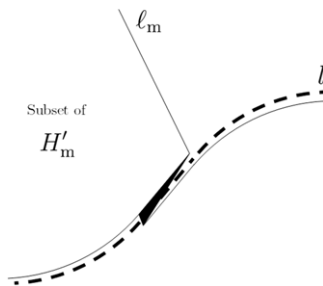


Fig. 23. Color dividers. The flipped-over portion of the paper is the part with its black face upwards; it is bounded by creases ℓ_m and ℓ_v . Both the flipped-over portion and ℓ_v are hidden underneath H'_m ; for the sake of a clear presentation, both of them are not labeled.

appear on the external surface and white areas may appear on the internal surface. This problem can be solved by choosing shapes of bounded paper which will not produce these black areas. Such shapes are identified by the following theorem.

Theorem 4. *If f is an isometric and noncrossing map from the xy plane to the folded model of a rotationally symmetric polyhedron, and the set of all simple closed curves in the xy plane that enclose the extruded polyhedron in the folded model is L , then every element in $\{f^{-1}[l] : l \in L\}$ is the boundary of a piece of paper that is a color divider of the model.*

Proof. Given some $l \in L$, consider the external surface of the model. If the parts of the model outside l were removed and the model was unfolded, the boundary of the unfolded paper would be $f^{-1}[l]$. For any pleat in the folded model, the subset of H'_m that has not been removed terminates exactly at l , just like the flipped-over region between ℓ_m and ℓ_v (Fig. 23). Hence, the black flipped-over region will be entirely concealed beneath the remaining white region H'_m . A similar argument applies to the internal surface and it can be shown easily that no white regions will be visible. \square

7. Limitations and future work

While our algorithm can only fold rotationally symmetric polyhedra by composing right frusta, we plan to extend the algorithm such that it can fold a larger class of polyhedra that includes asymmetric solids. This may be possible by investigating the extrusion of asymmetric frusta and composing them together. Also, not all of the solids generated by rotating the profile can be folded; for instance, the profile cannot self-intersect. In any profile, no vertex that is at the base of a positive frustum can lie on the axis of rotation, otherwise the paper has to be folded into a

single point to pass through that vertex, which is impossible with a finite number of folds. Intersections between the internal flaps of positive frusta can be resolved with the method presented in Lemma 2, but intersections between the internal flaps of negative frusta and the surrounding parts of the model have not been considered. Those flaps are planar and highly mobile, so they can generally be folded to avoid intersections, and possible future work includes an explicit algorithm to resolve such intersections. The possibility of applying our work to nanofabrication is brought closer by recent success in simulation of graphene folding using water nanodroplets [11] and experimental success in origami nanofabrication [12]. However, limitations such as a radius of curvature in folds and the presence of internal flaps in models folded from our algorithm need to be investigated further for this application.

8. Conclusion

We have derived a new algorithm to draw the crease patterns of rotationally symmetric polyhedra (Algorithm 1), which operates by decomposing the polyhedron into right frusta and composing their crease patterns (Algorithm 2). These algorithms have been implemented by an original CAD program that allows users to specify the profile and order of symmetry of their target polyhedron and generate the crease pattern automatically. This algorithm solves the problem present in [8–10] work on folding rotationally symmetric solids of external flaps which do not conform to the polyhedron's outer surface. The paper surrounding our extruded polyhedron is flat, hence it can be used to fold other origami models. This advantage allows the polyhedron to be extruded from a flat surface of any origami model. The algorithm allows the deliberate design of three-dimensional structures, which is suitable for use in industrial packaging and may also be applicable to nanofabrication in the future.

Acknowledgements

The authors would like to thank the editors and anonymous reviewers for their constructive suggestions.

References

- [1] Demaine ED, O'Rourke J. Geometric folding algorithms: linkages, origami, polyhedra. New York: Cambridge University Press; 2007.
- [2] Lang RJ. A computational algorithm for origami design. In: Proc. twelfth annual symposium on Computational geometry. 1996. p. 98–105.

- [3] Lang RJ. *Origami design secrets: mathematical methods for an ancient art*. Alamo: CRC Press; 2011.
- [4] Tachi T. Origamizing polyhedral surfaces. *IEEE Transactions on Visualization and Computer Graphics* 2009;99:298–311.
- [5] Demaine ED, Demaine ML, Mitchell JSB. Folding flat silhouettes and wrapping polyhedral packages: new results in computational origami. *Computational Geometry* 2000;16:3–21.
- [6] Benbernou NM, Demaine ED, Demaine ML, Ovadya A. Universal hinge patterns to fold orthogonal shapes. In: *Proc. fifth international meeting of origami science, mathematics, and education*. 2011. p. 405–21.
- [7] Demaine ED, Demaine ML, Ku J. Folding any orthogonal maze. In: *Proc. fifth international meeting of origami science, mathematics, and education*. 2011. p. 449–55.
- [8] Mitani J. A designing method for axial symmetrical curved origami with triangular prism protrusions. In: *Proc. fifth international meeting of origami science, mathematics, and education*. 2011. p. 437–48.
- [9] Mitani J. A design method for 3D origami based on rotational sweep. *Computer-Aided Design and Applications* 2009;6:69–79.
- [10] Lang RJ. Origami flanged pots from the Wolfram Demonstrations Project. <http://demonstrations.wolfram.com/OrigamiFlangedPots/> [accessed September 2011].
- [11] Patra N, Wang B, Král P. Nanodroplet activated and guided folding of graphene nanostructures. *Nano Letters* 2009;9:3766–71.
- [12] In HJ. Origami nanofabrication of three-dimensional electrochemical energy storage devices. Ph.D. Thesis. Cambridge: Massachusetts Institute of Technology, MIT, Department of Mechanical Engineering; 2005.