Folding Flat Silhouettes and Wrapping Polyhedral Packages: New Results in Computational Origami ⋆

Erik D. Demaine and Martin L. Demaine

Department of Computer Science, University of Waterloo, Waterloo, Ontario N2L 3G1, Canada, email: {`eddemaine, mldemaine`}`@uwaterloo.ca`.

Joseph S. B. Mitchell

Department of Applied Mathematics and Statistics, State University of New York, Stony Brook, NY 11794-3600, email: `jsbm@ams.sunysb.edu`.

Abstract

We show a remarkable fact about folding paper: From a single rectangular sheet of paper, one can fold it into a flat origami that takes the (scaled) shape of *any* connected polygonal region, even if it has holes. This resolves a long-standing open problem in origami design. Our proof is constructive, utilizing tools of computational geometry, resulting in efficient algorithms for achieving the target silhouette.

We show further that if the paper has a different color on each side, we can form any connected polygonal pattern of two colors. Our results apply also to polyhedral surfaces, showing that any polyhedron can be "wrapped" by folding a strip of paper around it. We give three methods for solving these problems: the first uses a thin strip whose area is arbitrarily close to optimal; the second allows wider strips to be used; and the third varies the strip width to optimize the number or length of visible "seams" subject to some restrictions.

Key words: paper folding, origami design, polyhedra, polyhedral surfaces, Hamiltonian triangulation, straight skeleton, convex decomposition

1 Introduction

Origami provides a rich field of research questions in geometry. At the ACM Symposium on Computational Geometry in 1996, Robert Lang's popular talk [17]

⋆ A preliminary version of this paper appears in the *Proceedings of the 15th Annual ACM Symposium on Computational Geometry* (Miami, Florida, 1999) 105–114.

helped to introduce the computational geometry community to this exciting area of research.

A classic open question in origami mathematics is whether every simple polygon is the silhouette of a flat origami. This question was first formally stated within the algorithms community by Bern and Hayes at the ACM-SIAM Symposium on Discrete Algorithms in 1996 [7]. More generally, we might ask whether every polygonal region (polygon with holes) is the silhouette of some flat origami. In this paper, we show that the answer is yes, and we provide constructive methods for achieving such origamis.

A more general problem in origami design is to take a sheet of *bicolor* paper, having a different color on each side, and fold it into a desired pattern of two colors. For example, John Montroll's book *Origami Inside-Out* [21] is entirely about such models. Taichiro Hasegawa [11] has designed an entire alphabet, including lower- and upper-case letters as well as punctuation. One origami designer, Toshikazu Kawasaki, has looked at the special case of *iso-area foldings*, that is, foldings that use equal amounts of both colors [12, pp. 96–97] [13, pp. 26–34]. See Fig. 1.

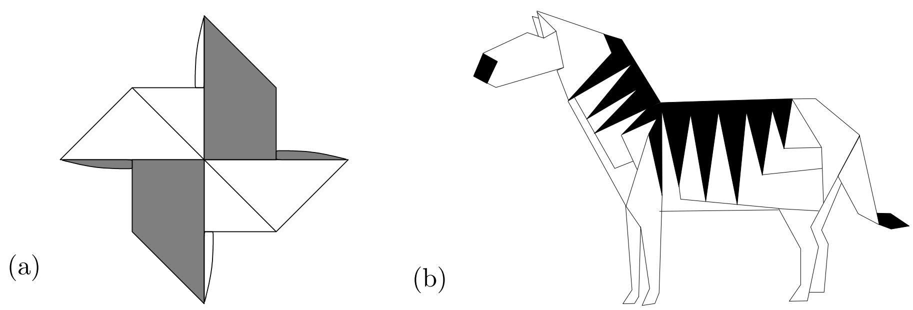

Fig. 1. (a) Iso-area pinwheel from [12, p. 97]. (b) Zebra by John Montroll from [20, pp. 94–103].

Formally, we define a *polygonal pattern* $\mathcal{P}$ to be a 2-colored polygonal subdivision of a polygonal region, each subregion of which may have holes. Our most general flat origami question then asks if there exists a flat folding of a sufficiently large piece of bicolor paper such that the top side of the flat origami gives exactly the input 2-color pattern, $\mathcal{P}$.

A more general question asks whether every polyhedron can be *wrapped* with a piece of rectangular paper. This is motivated not only by the problem of constructing three-dimensional origamis, but also the "gift wrapping problem," which was introduced to us by Jin Akiyama [3,4]. We define a *polyhedron* $\mathcal{P}$ very generally to be any connected union of pairwise-interior-disjoint polygonal regions (called faces), each of which lies on a plane in 3-space; we let n denote the number of vertices of $\mathcal{P}$. We also consider polyhedra whose faces are 2-colored. We then ask: Is every polyhedron $\mathcal{P}$ the folding of some sufficiently large rectangular piece of paper? If so, is there a folding of a bicolor sheet of paper that respects the face coloring? We answer both questions in the affirmative with one of our main results:

Theorem 1 *Given any (nonconvex) polyhedron, with each face assigned one of two colors, there is a folding of a sufficiently large square of bicolor paper that folds into the polyhedron with the desired colors showing on each face. An implicit representation of such a folding can be computed in time polynomial in n. The folding requires a number of folds polynomial in n and the ratio of the maximum diameter of a face of $\mathcal{P}$ to the minimum feature size of $\mathcal{P}$ (the smallest altitude of a triangle determined by any three vertices of any one face of $\mathcal{P}$).*

Note that a consequence of this theorem is that we can also achieve any *two-sided* 2-colored polygonal pattern, since we can treat it as a degenerate form of polyhedron.

In this paper, we give three methods for constructing flat origamis and polyhedral wrappings, resulting in constructive proofs of the above theorem. All three methods are based on the use of a *strip*, which is a rectangular sheet of paper, of width w. If our initial sheet of paper is given as a square (as is common in origami), then we can readily produce a strip from the square by a standard "accordion fold." Typically, we think of w as being relatively small, so that the strips are narrow (and an accordion fold from a square results in a thick strip of paper); however, we also consider the objective of having as wide a strip as possible.

In Section 2, we describe our main folding gadgets for strips, which we use throughout our constructions. Then, in Sections 3-5, we present our three methods, which differ in their objectives and their results:

- The "zig-zag" method (Section 3) utilizes a piece of paper whose area is arbitrarily close to optimal (the surface area of $\mathcal{P}$). It is based on the use of Hamiltonian triangulations.
- The "ring" method (Section 4) is based on straight skeletons and allows the widest possible constant-width strip.
- The convex-decomposition method (Section 5) is designed to create a folding with a desired pattern of visible "seams" that has convex faces between seams.

Throughout the paper, our figures follow standard origami conventions for depicting folds and seams: "mountain" folds are denoted with dash-dotted segments, "valley" folds are denoted with dashed segments, hidden edges and folds ("x-ray lines") are shown dotted, and visible seams and edges are shown solid.

2 Gadgets

We describe three fundamental gadgets that we use in our constructions: "hiding excess paper" with respect to a convex silhouette, "turning" a strip at a desired angle θ, and reversing the color of the top side of a strip.

2.1 Hiding Excess Paper

Convex polygons seem to be one of the few classes of polygons that are easy to make as silhouettes. The basic idea is to place a piece of paper P on top of the desired convex polygon Q so that P covers Q, and then fold the excess paper $P-Q$ along each edge of Q until it lies within Q. More generally, we will frequently make a folding that covers some desired region, but also has excess paper outside the region. Thus, an important operation is that of folding the excess paper underneath the region, thereby *hiding* the excess.

Let us make the idea of hiding more formal; refer to Fig. 2. Suppose we have a polygonal portion P of the paper that is joined to the rest of the paper along an edge e of P. Also joined to this edge e is a convex region C of paper underneath of which we are allowed to "hide" P. We need not assume that P covers C, in general. We let ρ_C denote the minimum feature size of C, defined to be the minimum distance between two vertices of C or between a vertex of C and an edge not incident to the vertex. We let δ_P denote the diameter of polygon P.

The *hide gadget* must fold P so that the excess paper, $P-C$, lies underneath C; it can fold P arbitrarily, provided that e does not move during the folding (otherwise, the folding would affect the rest of the paper).

Theorem 2 *There is a finite algorithm to compute a hide gadget, which uses* $O(|C|\log(\delta_P/\rho_C)+1/\theta_{min})$ *folds, where* $\theta_{min}=\min_i\theta_i$ *is the minimum among the interior angles* θ_i *at vertices of* C*, and* $|C|$ *is the number of vertices of* C*.*

PROOF. Let $e_0=e, e_1,\dots,e_k$ be the edges of C, in clockwise order. Let θ_i be the interior angle at vertex $v_i=e_i\cap e_{i+1}$. Let ℓ_i denote the line containing edge e_i. The two lines ℓ_i and ℓ_{i+1} cross at v_i, forming four cones, which we denote by $K_i^{(1)}$, $K_i^{(2)}$, $K_i^{(3)}$, $K_i^{(4)}$, where $K_i^{(1)}$ is the cone that lies locally within C, and the indices proceed counterclockwise about v_i. Refer to Fig. 2.

There are at most three "sharp" vertices (having corresponding interior angle less than $\pi/2$) of any convex polygon C; we let s_1, s_2, and s_3 denote the indices of such vertices (v_{s_1}, v_{s_2}, and v_{s_3}), if they exist. It is convenient to consider

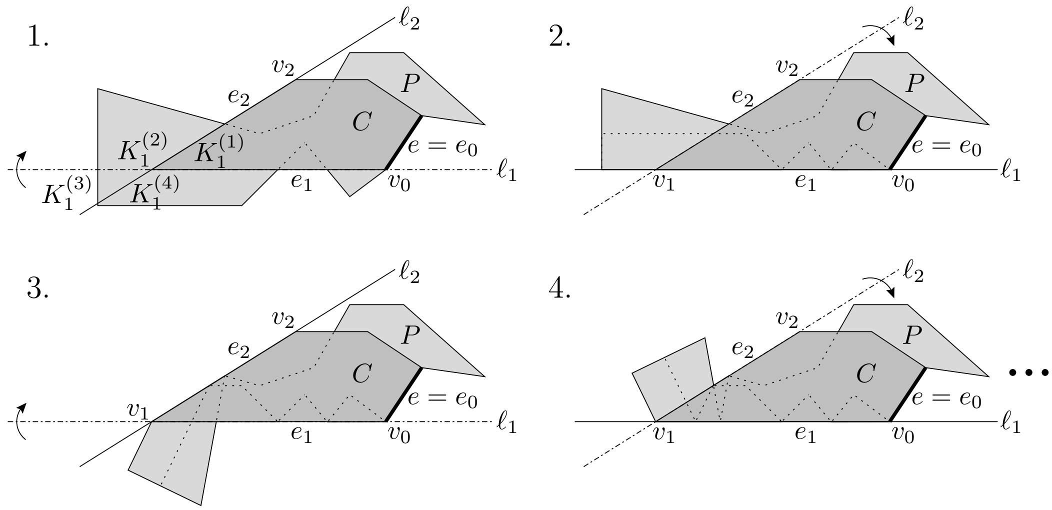

Fig. 2. An illustration of the hide gadget. 1. Hide polygon P underneath C, showing the notation in the proof of Theorem 2. 2-4. Results of the first three folds.

C as having zero-length "edges" at the sharp vertices, with an orientation orthogonal to the bisector of the corresponding interior angle. Then, with the consideration of these "sharp edges," interior angles are now effectively all greater than $\pi/2$.

We let $\epsilon = \rho_C/10$ and let $C_{+\epsilon}$ denote the "fattened" convex polygon defined by the intersection of the halfspaces that result from offsetting each of the lines ℓ_i through edges (and sharp edges) of C *outward* by an amount ϵ. Similarly, let $C_{-\epsilon}$ denote the "shrunk" convex polygon defined by the intersection of the halfspaces that result from offsetting each of the lines ℓ_i *inward* by an amount ϵ. (Our choice of $\epsilon < \rho_C$ guarantees that $C_{-\epsilon}$ exists, is nonempty, and has the same number of edges as does C.) Note that $C_{-\epsilon} \subset C \subset C_{+\epsilon}$ and that $C_{+\epsilon}$ has no sharp vertices (while C and $C_{-\epsilon}$ may have sharp vertices).

Our first goal is to fold the excess paper of P in such a way that the paper lies within $C_{+\epsilon}$. This is readily done using $O(|C| \log(\delta_P/\rho_C))$ folds, by folding along the lines parallel to the edges of $C_{+\epsilon}$, in order around $C_{+\epsilon}$, first at a distance $d/2$ from the corresponding edge (or vertex that defines a sharp edge) of C, then at distance $d/4$, $d/8$, etc., where $d \leq \delta_P$ is the distance by which the excess paper of P originally extends outside the corresponding edge of C. It is important to note that, because any two consecutive edges of $C_{+\epsilon}$ define an interior angle greater than $\pi/2$, the folding that we do in order to get all of the excess within distance ϵ of some edge (or sharp vertex) of C is not "undone" by the folds that we do parallel to the succeeding edge: folds parallel to e_{i+1} will only bring points closer also to edge e_i.

Having all excess paper within $C_{+\epsilon}$, we proceed to fold the excess under C using mountain folds along the lines ℓ_i that define (non-sharp) edges of C, in a manner we are about to describe. During this process, any point p on the excess paper that we fold under along ℓ_i must stay within distance ϵ of ℓ_i: the segment joining p to the closest point on ℓ_i gets mapped to a polygonal chain, of the same length as the segment, after any number of folds, implying that p must stay within distance ϵ of ℓ_i. In particular, point p on the excess that is folded under along ℓ_i cannot be mapped by the fold to a point that is outside an edge $e_j \neq e_{i+1}, e_{i-1}$. (Our choice of ϵ being substantially smaller than the minimum feature size of C guarantees that points in the excess in the ϵ-neighborhood of v_i stay within this ϵ-neighborhood, never entering the ϵ-neighborhood of some other vertex, under any foldings along ℓ_i and ℓ_{i+1}.)

We begin by folding under the excess of P by using a mountain fold along ℓ_1. Now, in the neighborhood of v_1, the excess lies entirely in the cone $K_1^{(2)}$, of angle $\pi - \theta_1$. Next we fold under the excess with a mountain fold along ℓ_2. If the angle θ_1 is at least $\pi/2$, there will be no excess paper of P in the neighborhood of v_1 (all excess lies in the cone $K_1^{(1)}$ and is hidden by C in the neighborhood of v_1), and we continue by folding under along ℓ_3, then ℓ_4, etc., until the first sharp vertex, say v_{s_1}. (In Fig. 2, $v_{s_1} = v_1$.) After the fold along ℓ_{s_1}, there may be excess paper of P in the neighborhood of v_{s_1} in the cone $K_{s_1}^{(4)}$, but it now occupies a cone, within $K_{s_1}^{(4)}$, having angle at most $\pi - 2\theta_{s_1}$. After another fold under along ℓ_{s_1}, any excess lies within a cone of angle at most $\pi - 3\theta_{s_1}$, within $K_{s_1}^{(2)}$. Each successive folding under of excess paper (alternating between ℓ_{s_1} and ℓ_{s_1+1}) decreases the angle by another θ_{s_1}, so this process must terminate in at most $1 + (\pi - \theta_{s_1})/\theta_{s_1}$ steps. Finally, there is no excess in cones $K_{s_1}^{(2)}$, $K_{s_1}^{(3)}$, or $K_{s_1}^{(4)}$; any excess paper of P lies in cone $K_{s_1}^{(1)}$, and is locally covered by C.

Having completed the hiding of paper at v_{s_1}, we then proceed to folding under the excess along ℓ_{s_1+1}, ℓ_{s_1+2}, etc., until we encounter the next sharp vertex v_{s_2}, at which point we again may require multiple foldings in order to hide the excess paper in the neighborhood of v_{s_2}. We may then have to proceed to a third sharp vertex before completing our foldings around C. Since there are at most three vertices at which we need multiple ($O(1/\theta_{min})$, where $\theta_{min} = \min_i \theta_i$) folds, we obtain the claimed overall bound on the number of folds of the hide gadget. □

An immediate consequence of this theorem is the following:

Corollary 3 *Given any polygon P and convex polygon Q, such that P can be moved to cover Q, P can be folded into a flat origami whose silhouette is Q, using $O(|C| + 1/\theta_{min})$ folds.*

2.2 Turning a Strip

A natural tool to fold a paper strip into a desired shape is the ability to *turn* the strip. More formally, we will consider turns of the following sort. Take two infinite strips S and T in the plane, and consider their intersection $I = S \cap T$; see Fig. 3. Label the two connected regions of $S - T$ (respectively, $T - S$) by S_1 and S_2 (respectively, T_1 and T_2). The *turn gadget* must fold a strip so that it covers precisely $U = S_1 \cup I \cup T_1$.

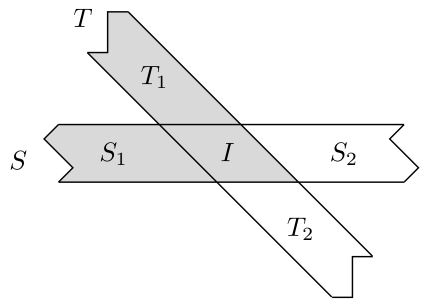

Fig. 3. A turn must cover precisely two connected portions of $S - T$ and $T - S$ as well as $I = S \cap T$.

Our turn gadget is shown in Fig. 4. The first fold is perpendicular to the edges of S and is incident to the convex vertex of U. The second fold is an angular bisector of the convex angle θ, effecting the turn. If $\theta \geq \pi/2$, these two folds are all that are needed. On the other hand, if $\theta < \pi/2$, they leave a right-angle triangle of excess paper, whose angle incident to the convex vertex is $\pi/2 - \theta$. We can hide this triangle underneath U by "wrapping" it around the angle θ, which requires

$$\left\lceil \frac{\pi/2 - \theta}{\theta} \right\rceil = \left\lceil \frac{\pi}{2\theta} \right\rceil - 1$$

extra folds.

We have thus proved the following lemma.

Lemma 4 *Given two strips S and T in the plane, and given any connected region S_1 (respectively, T_1) of $S - T$ (respectively, $T - S$), a strip can be folded into a flat origami whose silhouette is precisely $S_1 \cup (S \cap T) \cup T_1$.*

It turns out that if we apply a sequence of turn gadgets, the first fold of a particular turn gadget (which involves folding through all layers) may destroy the effect of previous turn gadgets, that is, uncover regions that were covered by previous turn gadgets. This can be avoided by using a *generalized turn gadget*, which involves letting the strip go past the turn, making the perpendicular fold once it has gone far enough to avoid destruction, and then bringing the

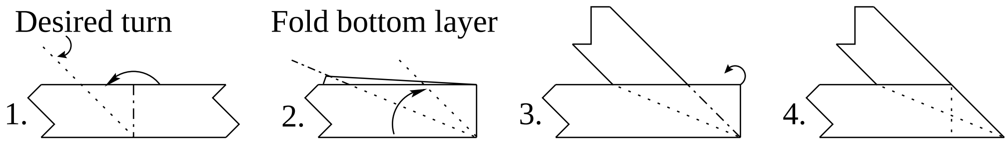

Fig. 4. Folding a turn gadget. Step 3 hides the excess paper and is only necessary for $\theta < \pi/2$.

strip back before making the second (turning) fold. See Fig. 5. We now obtain a trapezoid of excess of paper, which can be folded underneath by applying Theorem 2.

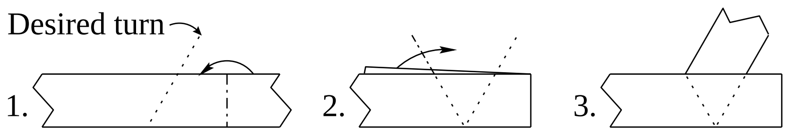

Fig. 5. Folding a generalized turn gadget.

Generalized turn gadgets will also be important to produce useful overhang, as we will see in Section 3.

2.3 Color Reversal

We utilize a *color-reversal gadget*, as shown in Fig. 6. It consists of three folds: a perpendicular fold, and two 45° folds. The result is a color reversal (that is, an exchange of the showing side of the strip) along the perpendicular edge. Note that the triangle of excess paper underneath the finished gadget can, if necessary, be reduced in size by the gadget of Theorem 2.

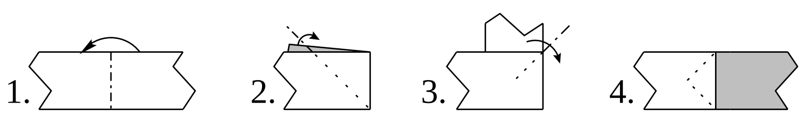

Fig. 6. Folding a color-reversal gadget.

3 Zig-Zag Method

Our first method of folding a strip into a desired polyhedron $\mathcal{P}$ is based on a Hamiltonian triangulation of P's surface. A triangulation is *Hamiltonian* if its dual graph is Hamiltonian: there is a path that visits each node (triangle) exactly once. We find such a triangulation by first computing a triangulation $\mathcal{T}$ of the faces of $\mathcal{P}$, and then finding a Hamiltonian refinement of $\mathcal{T}$. A *Hamiltonian refinement* of $\mathcal{T}$ is a Hamiltonian triangulation obtained from $\mathcal{T}$ by partitioning each of its triangles into one or more subtriangles, each of which inherits the color of the containing triangle. Arkin et al. [6] have shown that any connected triangulation $\mathcal{T}$ has a Hamiltonian refinement; see Fig. 7.

Our algorithm begins by computing a Hamiltonian refinement, $\mathcal{T}'$, of $\mathcal{T}$, of $O(n)$ subtriangles, along with an associated Hamiltonian path, γ. This requires $O(n)$ time, since it involves nothing more than a spanning tree (e.g., depth-first search tree) computation in the dual graph of $\mathcal{T}$. (If we are not given a triangulation of the faces of $\mathcal{P}$, then we first compute $\mathcal{T}$, which is readily done in $O(n \log n)$ time.)

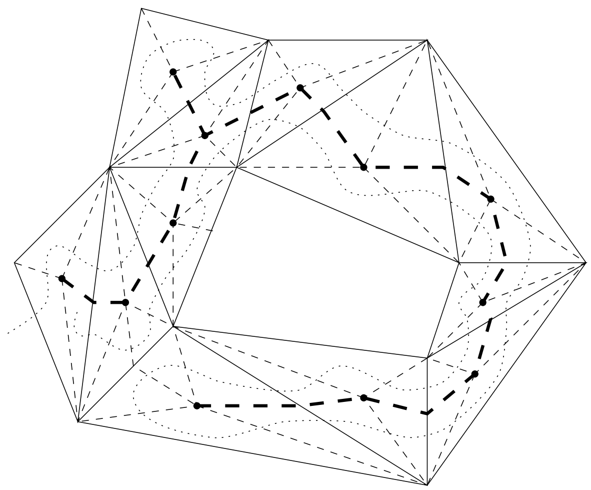

Fig. 7. Hamiltonian refinement of a triangulation $\mathcal{T}$ having 12 triangles (shown with solid edges). A Steiner point (shown as a solid circle) lies interior to each of the 12 triangles and is joined by dashed segments to each corner, and to the midpoint of each edge shared by two triangles. The dashed segments decompose the 12 triangles into subtriangles, for which a Hamiltonian path γ (dotted curve) is readily obtained by walking around a spanning tree (shown with heavy dashed segments).

We now traverse each triangle of the triangulation $\mathcal{T}'$, in the order prescribed by the Hamiltonian path γ. Let $T_1, \ldots, T_k$ be the sequence of $k = O(n)$ triangles along γ, with T_i sharing the edge e_i with T_{i+1}. Let v_i denote the vertex of T_i that is opposite edge e_i.

We cover the triangle T_i by zig-zagging the strip in rows parallel to edge e_i, starting with a row having one edge passing through vertex v_i; see Fig. 8. The zig-zagging is effected by "turn-arounds" that take place just beyond the boundary of T_i. A turn-around can be achieved using two consecutive right-angle turn gadgets, or simply by making two consecutive 45° folds, as shown in Fig. 9. Further, by adjusting the turn-around gadget slightly, as shown in the figure, we can make row j overlap partially with row $j+1$, allowing us to control the parity of the zig-zagging as well as the parallel shift of the rows. In this way, we can ensure that the final row in the coverage of T_i has one edge coinciding with e_i, while completing this row at the endpoint v_{i+1}.

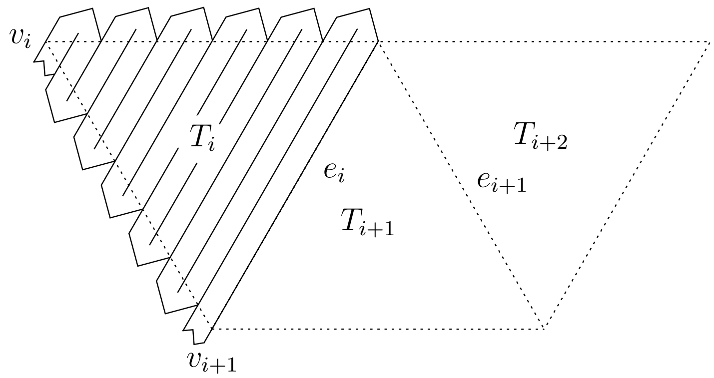

Fig. 8. We cover triangle T_i by zig-zagging in rows parallel to e_i, such that we end up at vertex v_{i+1}.

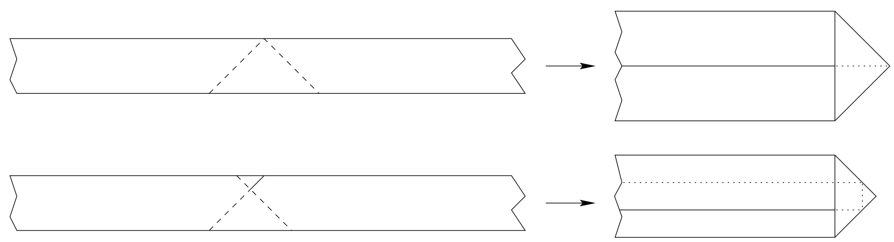

Fig. 9. Turning around when going from one row to the next, possibly with partial overlap between rows (bottom).

The zig-zagging coverage of T_i results in some excess paper spilling over the edges of T_i. This excess is readily folded under by Theorem 2. Further, the total area of paper used to cover T_i is bounded above by $area(T_i)$ plus the amount A_t of excess caused by turn-arounds, plus the amount A_o of excess caused by overlaps between rows, arising from the need for the last row to line up with edge e_i, while completing at v_{i+1}.

We obtain estimates of A_t and A_o as follows. Let h_i denote the altitude of T_i, given by the distance from v_i to the line containing e_i. Then, the zig-zag coverage of T_i can be accomplished using at most $m_i = \lceil (h_i/w) \rceil + 1$ rows, where the "+1" term arises from the possible extra row required to meet the parity constraint (to end at v_{i+1}). Each turn-around utilizes area at most $O(w^2 \cot \theta_{min})$, where θ_{min} denotes the smaller of the two interior angles of T_i at the endpoints of e_i; see Fig. 10. Thus, we have $A_t = O(h_i w \cot \theta_{min})$, which implies that $A_t = O(wL_i)$, where L_i is the length of the longest side of T_i. Also, we see that $A_o = O(w|e_i|) = O(wL_i)$, since the overlaps between rows need not consume more strip length than twice the longest row (which is roughly of length $|e_i|$). (We have *twice* the longest row because one extra row may be needed to compensate for the round-up from (h_i/w) to $\lceil (h_i/w) \rceil$, while a second extra row may be needed for the parity constraint.) We summarize with

Lemma 5 *The coverage of triangle T_i utilizes a strip of area at most $area(T_i)+$*

$O(wL_i)$, where w is the width of the strip and L_i is the length of the longest side of T_i.

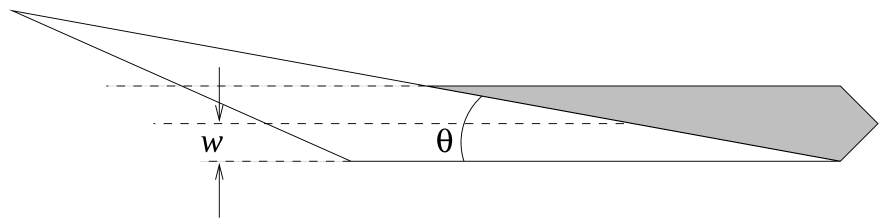

Fig. 10. Area estimate for excess paper (shown shaded) that spills over during a turn-around. The area of the shaded region is $w^2 + (2w)(2w)\cot\theta$.

The transition from triangle T_i to T_{i+1} involves turning the strip in such a way that the strip becomes parallel to the edge e_{i+1}, while creating excess that can be folded under $T_i \cup T_{i+1}$. Refer to Fig. 12. This could be done using the generalized turn gadget of Section 2.2, but for turn angles of more than $\pi/2$, the amount of excess paper is too large: it grows arbitrary large as the turn angle approaches π. In this case, we use an alternate turn gadget shown in Fig. 11. Note that this turn gadget solves a different problem from the one described in Section 2.2 (the corner does not have to be "filled in"), which allows us to reduce the amount of excess paper to $O(w^2)$.

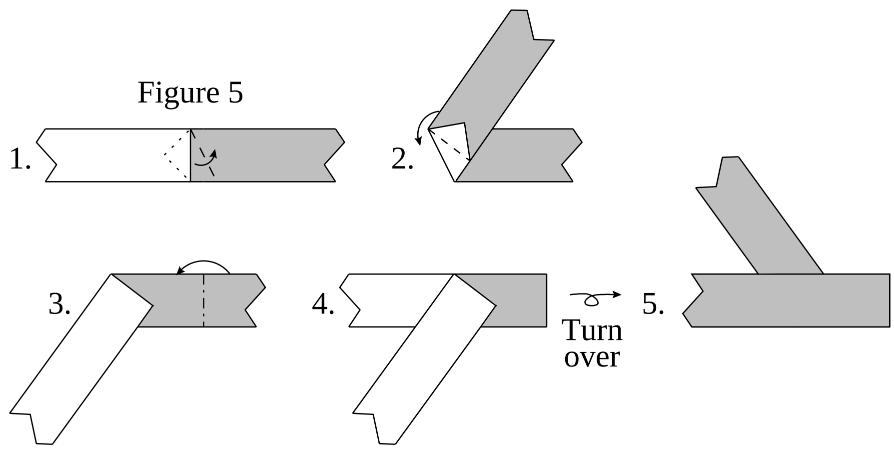

Fig. 11. Folding an alternate turn gadget, which reduces the amount of excess paper for turn angles of more than $\pi/2$. Step 3 can be adjusted to produce the desired amount of overhang.

By induction, we can cover all the triangles of $\mathcal{T}'$ (and hence of $\mathcal{T}$) in this way. Note that we can also change which side of the strip is up, as we make the transition between two triangles, using the gadget in Fig. 6. Thus, we can control the coverage in such a way that we preserve a given 2-coloring of $\mathcal{T}'$ (which is inherited from a 2-coloring of $\mathcal{T}$ or of the original faces of $\mathcal{P}$).

Since the transition from triangle to triangle uses at most $O(wL)$ excess paper area, where $L = \max_i L_i$, Lemma 5 applied to the $O(n)$ triangles in turn yields the following result:

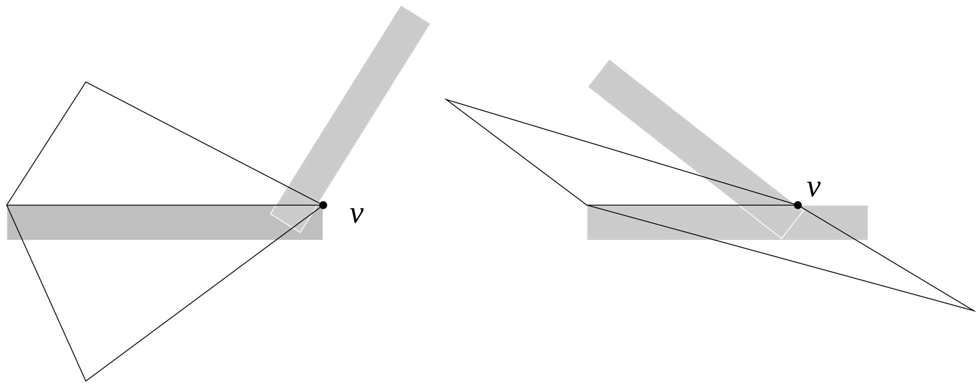

Fig. 12. Turning from one triangle to another. Note that the turn must have some overhang to finish covering the triangle.

Lemma 6 *The coverage of $\mathcal{T}$ requires a strip of area at most area$(\mathcal{T})$ + $O(nwL)$.*

Using this zig-zag method with sufficiently narrow strips ($w \to 0$), we obtain, as a consequence of Lemma 6, the following result on optimal paper usage.

Theorem 7 *Let A be the surface area of a given 2-colored polyhedron. Then for any $\epsilon > 0$, there is a rectangle R of bicolor paper with area at most $A + \epsilon$ such that R folds into the polyhedron with the desired colors showing.*

Remark. Instead of using a very small width w, our approach also allows one to use a strip with a larger width, up to the smallest altitude of the triangles in $\mathcal{T}'$. Of course, this increases the excess paper that needs to be folded under, and increases the total area of paper required, while decreasing the number of "seams." In Section 5 we discuss seams and their minimization via our third method, base on convex decompositions.

Finally, we conclude that the zig-zag method gives us our first proof of Theorem 1. The time required for the method to produce an implicit representation of the folding is simply $O(n)$ (after triangulation of the faces of $\mathcal{P}$, which takes time no more than $O(n \log n)$, even for multiply connected faces). The actual folding produced by the method utilizes $O(n)$ gadgets, each of which requires $O(1/\theta_{min})$ folds, where θ_{min} is the smallest angle of any triangle in the Hamiltonian triangulation $\mathcal{T}'$. (Here, the hide gadget is applied with $|C| = O(1)$ and $\log(\delta_P/\rho_C) = O(\log(1/\theta_{min}))$.) Since θ_{min} can readily be bounded within a constant factor of the feature size (as defined in Theorem 1), we obtain the bound on the number of folds as stated in the theorem.

4 Ring Method

Our second method is based on covering a polyhedron by a collection of "rings." This method's main advantage is that it allows the strip to have the largest possible width in the case that the strip width is not allowed to change.

This section is outlined as follows. Section 4.1 defines rings and shows how to cover a polyhedron with rings. Section 4.2 gives an outline of the algorithm. Section 4.3 fills in the details by showing how to fold rings and how to bridge between rings. Finally, Section 4.4 analyzes the requirements on the strip width.

4.1 Rings and the Straight Skeleton

We define a *ring* to be a pair of (potentially non-simple) polygons, called *walls*, such that "shrinking" one wall results in the other. *Shrinking* consists of continuously insetting each boundary vertex, so that at any particular time, every shrunken boundary edge is parallel to the original, and the perpendicular distance between the shrunken and original boundary edges is the same for all boundary edges. Conceptually, the walls are parallel to each other except at turns, and have constant "width" all around. Some examples of rings are given in Fig. 13; note that the walls are allowed to be non-simple polygons.

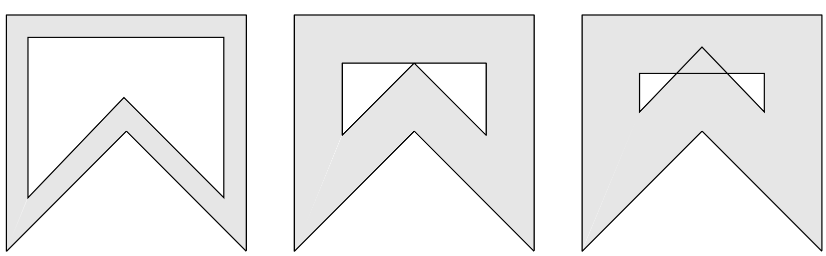

Fig. 13. Examples of rings.

The idea of rings is based on a different shrinking process that is used in the definition of the *straight skeleton* [1]. To distinguish between the two shrinking processes, we call the one described above "topological shrinking," because it ignores non-simplicities as the polygon shrinks. In contrast, *geometric shrinking* stops whenever the polygon becomes non-simple, and instead recursively shrinks each subregion. The straight skeleton of a polygonal region is defined to be the union of the line segments along which the boundary vertices travel, as we geometrically shrink the polygonal region. See Fig. 14 for an example.

While we are not actually interested in the straight skeleton, we are interested in the *ring decomposition* induced by it. During the geometric shrinking pro-

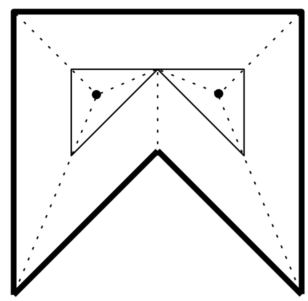

Fig. 14. Illustration of the straight skeleton (dotted lines) and skeleton walls (solid lines).

cess, the polygonal region changes in topology at finitely many times. At these times, we say that the boundary is a *skeleton wall*, and call the regions between these walls *skeleton rings*; see Fig. 14. Note that skeleton rings are indeed rings (by definition) and furthermore they partition the desired polygonal region. We can also cover a 2-colored polyhedron by rings, using the skeleton rings from each face.

The term "straight skeleton" was first coined by Aichholzer et al. [2], although the idea goes back to at least 1984 [19, pp. 98–101]. It was first defined for polygonal regions by Aichholzer and Aurenhammer [1], who give an $O(n^2 \log n)$-time algorithm for computing it. Recently, Eppstein and Erickson [9] developed an $O(n^{17/11+\epsilon})$-time algorithm. The straight skeleton has complexity $O(n)$.

Lemma 8 *The ring decomposition induced by a straight skeleton of a polygonal region P can be computed in time proportional to the total complexity of the subdivision (which is at most $O(n^2)$).*

PROOF. Each vertex v_i of P is incident to exactly one edge, $e_i = (v_i, u_i)$, with endpoint u_i interior to P at distance d_i from ∂P, the boundary of P. In time $O(n)$ we easily find $\min_i d_i$; assume, without loss of generality, that $d_1 = \min_i d_i$. Then, the outermost ring in the ring decomposition of P is bounded by two walls: ∂P and a shrunken version, q, of ∂P. Now, q passes through the endpoint u_1 and has one vertex on each of the edges e_i; the point u_1 is a "pinch off" point (cut vertex) of q. (In general, there may be multiple such pinch off points, if there are ties in determining $\min_i d_i$.) We readily construct q in time proportional to its complexity ($O(n)$), and then recurse within each of the rings enclosed by q. It is easy to see that this can be done in overall time that is proportional to the output size. □

4.2 Outline

The ring algorithm works as follows. We compute the skeleton rings in each face of the 2-colored polyhedron. Define the *skeleton dual* to be the graph with vertices corresponding to skeleton rings, and edges between two vertices corresponding to rings that share a wall edge. By the assumption that the polyhedron's surface is connected, the skeleton dual is connected, and hence we can find a spanning tree.

We use this spanning tree as a road map for our construction. We perform a depth-first traversal of the tree, starting from an arbitrary root. At each new node we traverse, we construct the corresponding ring. When we traverse a node that we have visited before, we can "walk" around the ring (by constructing part of it) and bring the strip to the desired joining place for an adjacent ring. Hence, we only need to show how to construct a skeleton ring, and how to connect between two skeleton rings with an optional color change.

4.3 Strip Rings

Instead of folding skeleton rings directly, we will cover them by a collection of *strip rings*, that is, rings with the same width as the strip. Strip rings are particularly attractive because they can be constructed simply by folding a sequence of generalized turn gadgets from Section 2.2. (We use *generalized* turn gadgets so that they do not interfere with each other.)

Lemma 9 *Given any ring R of width $|R|$ and a strip of width w, R can be covered by $\lceil |R|/w \rceil$ strip rings, each of which is contained in the current polygonal region.*

PROOF. Assume first that $|R| \geq w$. Then one way to build such a cover is as follows. Let $R = (q_0, q')$ be the ring (between walls q_0 and q') that we want to cover. In general, suppose we want to cover a ring $R_i = (q_i, q')$ such that $|R_i| \geq w$, for $i = 0, 1, \ldots$. Shrink or expand the wall q_i to pull it towards the interior of R_i by a perpendicular distance of w. The result is another wall q_{i+1} that is in R_i. Indeed, (q_i, q_{i+1}) is a strip ring.

It remains to cover the subring $R_{i+1} = (q_{i+1}, q')$ of R_i. If $|R_{i+1}| \geq w$, we can recursively apply this procedure. Each iteration decreases the width of the ring to cover by the constant w. Hence, after $k = \lfloor |R|/w \rfloor$ iterations, we are left with a strip $R_k = (q_k, q')$ whose width is less than w. If its width is zero (that is, w evenly divides $|R|$), we stop. Otherwise, we shrink or expand q' to pull it towards the interior of R_k, resulting in a wall q that is outside R_k but inside R. This last strip ring (q, q'), which contains R_k, completes the cover using $\lceil |R|/w \rceil$ strip rings.

Now assume that $|R| < w$, and let $R = (q_1, q_2)$. Consider topologically shrinking or expanding q_1 and q_2 to push them away from R, stopping when we find a ring R' that has the same width as the strip. If a wall hits the boundary of the polygonal region, we stop shrinking/expanding it. Because of the upper bound on the strip's width described in Section 4.4, we cannot have both walls hitting the boundary of the polygonal region. Hence, we obtain a strip ring R' that contains R and is contained in the polygonal region, the desired result. □

It only remains to show how to bridge between two strip rings. Specifically, we need to show how to combine strip rings in two different ways: between overlapping strip rings, and between touching strip rings possibly of different colors. In both cases, we take an arbitrary edge shared by the two strip rings; for overlapping rings, this "edge" has some thickness. We bridge at any joining place along this edge by using the turn-around gadget in Fig. 9. The excess paper can be reduced to fit within the two rings by applying Theorem 2. We can also reverse the color of the strip in between the two folds of the turn-around gadget (note that if the two rings have different colors, they do not overlap), using the color-reversal gadget in Section 2.3.

4.4 Strip Width

What are the least possible constraints on the strip's width? If our only building blocks are strip rings (in other words, the width of the strip stays essentially constant), we need the property that at least one strip ring fits inside the polygonal region we are trying to cover. One observation is that the strip's width must be at most the minimum feature size, that is, the minimum distance between two non-incident boundary edges. Indeed, we need a stronger upper bound on the strip's width than the minimum feature size, to ensure that it is possible to turn at every reflex vertex without falling outside the polygonal region.

Consider a reflex vertex v with exterior angle θ and consider the non-incident boundary edge e that is closest to v along the angular bisector at v; refer to Fig. 15. To turn at v, a ring turns at a point on the angular bisector of v. Let d denote the distance from v to e along the angular bisector of v. This gives us the maximum allowed "diagonal" width of the strip. This means that the true width of the strip must be at most $d \sin(\theta/2)$. By minimizing this expression over all reflex vertices, we obtain an upper bound on the strip's width for that face.

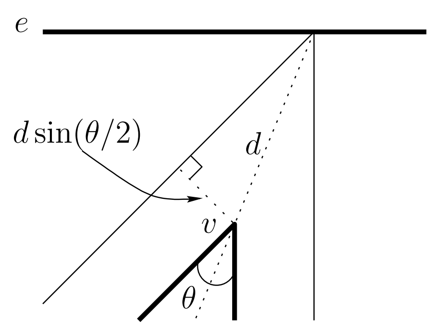

Fig. 15. Computing the upper bound on the width of the strip.

We choose the strip's width so that this upper bound is satisfied for every face of the polyhedron. Note that we can always thin the strip to achieve the desired upper bound by accordion folding it.

4.5 Summary

We conclude that the ring method gives us our second proof of Theorem 1. The time required for our method to produce an implicit representation of the folding is bounded by $O(n^2)$. The actual folding produced by the method utilizes $O(n^2)$ gadgets, each of which requires $O(1/\sigma)$ folds, where σ is the feature size of the ring decomposition. It is readily shown that the ring decomposition results in $1/\sigma$ being polynomially bounded in the feature size of the original input $\mathcal{P}$. Thus, we obtain the claimed bounds of Theorem 1.

5 Convex-Decomposition Method

The goal of our third method is to wrap a polyhedron while minimizing some metric involving "seams." A *seam* is a visible crease or edge of paper on the interior of a face of the polyhedron. There are several interesting metrics to optimize on seams. For example, minimizing

(1) the number of regions between the seams,
(2) the total number of seams, or
(3) the total length of the seams

may be relevant to finding a polyhedron wrapping with a pleasing exterior view.

Note that our method for covering a convex polygon (Corollary 3) uses no seams. Indeed, convex polygons are the maximal objects that can be folded without seams: because a reflex vertex has negative curvature (i.e., the sum of the incident angles, on both the top and bottom sides, is more than 2π), it cannot be folded without seams from a strip which has zero curvature everywhere (i.e., the sum of incident angles is exactly 2π).

However, this argument only applies when one seamless polygon is desired, as it is easy to make a seamless nonconvex region of a flat origami. The key is that the nonconvex vertices of a seamless region, which by themselves have negative curvature, can be made non-vertices, and hence have zero curvature, by adding seamless regions incident to the vertices.

For the purposes of this paper, we concentrate on the case in which the regions between seams are convex polygons. Hence, our third method is based on a *convex decomposition* of the polyhedron's surface, that is, a partitioning of each face of the polyhedron into interior-disjoint convex polygons. Each interior edge in the decomposition corresponds to a seam, and each polygon in the decomposition corresponds to a region between the seams. Thus, the three seam optimization questions stated above, subject to the convex-face restriction, can be rephrased as convex-decomposition questions:

(1) Decompose a polygonal region into the minimum number of convex polygons.
(2) Decompose a polygonal region into convex faces using the fewest number of edges.
(3) Decompose a polygonal region into convex faces using edges with minimum total length.

Depending on the kinds of seams we want to allow, we may or may not allow the addition of Steiner points. There are hence six questions of interest: the above three with and without Steiner points allowed. We know of no work explicitly addressing Problem 2. Note however that if Steiner points are disallowed, the number of edges is completely determined by the number of regions, and hence Problems 1 and 2 become the same.

There is an abundance of prior work on problems of convex decomposition; see the recent survey article of Keil [14]. For simple polygons (no holes), Greene [10] gives an $O(r^2n^2)$-time algorithm for Problems 1 and 3 without Steiner points (and hence also Problem 2 without Steiner points), where r is the number of reflex vertices. Keil [16] independently discovered an $O(r^2 n \log n)$-time algorithm for Problem 1 without Steiner points. The only polynomial-time algorithm for optimal convex decomposition with Steiner points is by Chazelle and Dobkin [8], who developed an $O(n+r^3)$-time algorithm for Problem 1 with Steiner points. Problems 2 and 3 with Steiner points appear to be interesting open problems.

Unfortunately, for polygonal regions with holes, these problems are all either known to be NP-hard, or seem likely to be. Lingas [18] showed that Problem 1 with Steiner points is NP-hard. Keil [15] showed that Problems 1 and 3 without Steiner points (and hence also Problem 2 without Steiner points) are NP-hard. Problems 2 and 3 with Steiner points again remain open, but are likely also NP-hard.

The method described in this section will conform to any convex decomposition of the polyhedron's surface, thereby proving the following theorem.

Theorem 10 *Given any convex decomposition of a 2-colored polyhedron's surface, there is a folding of a sufficiently large piece of bicolor paper that covers all the faces of the polyhedron with the assigned color, and has seams at precisely the edges of the convex decomposition.*

Depending on the choice of the convex decomposition, this method optimizes the desired metric of seams, subject to the convex-face restriction.

The method is based on dynamically adjusting the strip width, which is described in Section 5.1. This gadget is used in Section 5.2 to complete the convex-decomposition method.

5.1 Dynamic Strip Width

This section describes how the strip width can be changed along a perpendicular edge to have any width that is at most the original (physical) width.

The basic gadget is shown in Fig. 16. Note that the folding starts with the reverse side of the strip showing, and is flipped back over in Step 4. The first fold is the perpendicular along which we want to change the strip width, and is a valley from this orientation. The second fold is another perpendicular, which is the desired reduction amount away from the first fold. The third fold is a squash fold, which involves folding down the top part of the strip by the desired reduction amount, along a horizontal line; the upper-left corner naturally "squashes" along two 45-degree folds (which are originally right on top of each other). Equivalently, we can squash fold upwards the bottom part of the strip.

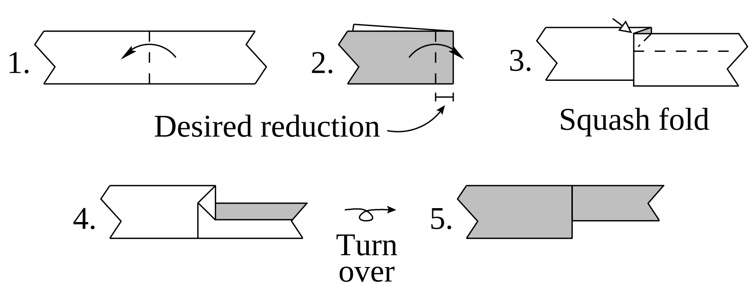

Fig. 16. Folding a strip-width gadget.

This gadget can reduce a strip of width w into a strip of width $\alpha \cdot w$ for any $\frac{1}{2} \leq \alpha \leq 1$. By applying the "reverse" of the gadget (that is, flipping the image horizontally), we can also undo any previous reduction. We are now ready to prove the desired theorem:

Theorem 11 *A strip can be repeatedly resized along various perpendicular edges to any width that is at most the original physical width. The number of folds required to change the width from w_1 to w_2 is*

$$O\Big(1 + |\log\ (w_1/w_2)|\Big).$$

PROOF. We maintain the invariant that the strip is the result of several width-halving gadgets (a strip-width gadget with $\alpha = \frac{1}{2}$), possibly followed by a general width-reduction gadget with some α. To achieve a particular strip width, we first fold (if necessary) the reverse strip-width gadget with the same α. Then we apply width-halving or reverse width-halving gadgets until the strip has width within a factor of two of the desired width. Finally, we apply the general width-reduction gadget to obtain the desired strip width. The bound on the number of folds follows immediately. □

Note that any excess paper from strip-width gadgets can be reduced to fit within any desired incident region (namely, the polyhedron face that we are covering), by Theorem 2.

5.2 Approach

We are now in the position to describe a folding that only has seams on the edges of a given convex decomposition of a polyhedron's surface. We define the *diameter* of the convex decomposition to be the largest diameter of any convex polygon in the decomposition, that is, the largest distance between any two points on a common convex polygon. We choose our strip to have this diameter as its physical (initial) width.

The algorithm works as follows. We traverse a spanning tree of the dual of the convex decomposition in a depth-first traversal. The strip always enters a convex polygon P along a sub-portion of one of its edges, perpendicular to that edge e. Reorient so that e is vertical. At this point of entry along e, we resize the strip to be the vertical extent of P. Note that the resized strip may not have the right vertical positioning to cover all of P; this can be fixed by using the *shift gadget* shown in Fig. 17.

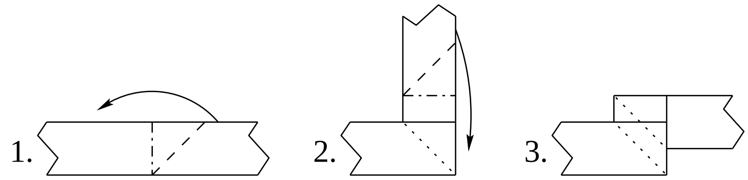

Fig. 17. Folding a shift gadget, which is just a sequence of two right-angle turn gadgets from Fig. 4.

Next we continue the strip straight until it covers all of P; we call this the *completion point*. Let e' denote the edge shared by P and the next polygon P' in the traversal order. If the length of e' is less than the current width of the strip (i.e., the vertical extent of P), then we resize the strip width at the completion point to the length of e'. It remains to show how to turn the strip to reach e' perpendicularly. In fact, this can be done using a generalized turn gadget (Section 2.2). If e' has negative slope, as in Fig. 18(a), the perpendicular fold is right at the completion point. If e' has positive slope, the perpendicular fold may be past the completion point, as in Fig. 18(b). In either case, the second fold turns onto the infinite strip perpendicular and incident to e'. Finally, any excess paper that results from the turn gadget extending beyond polygon P can be folded underneath P, by Theorem 2. (This does not cause any seams to appear, since the folding is done underneath P.)

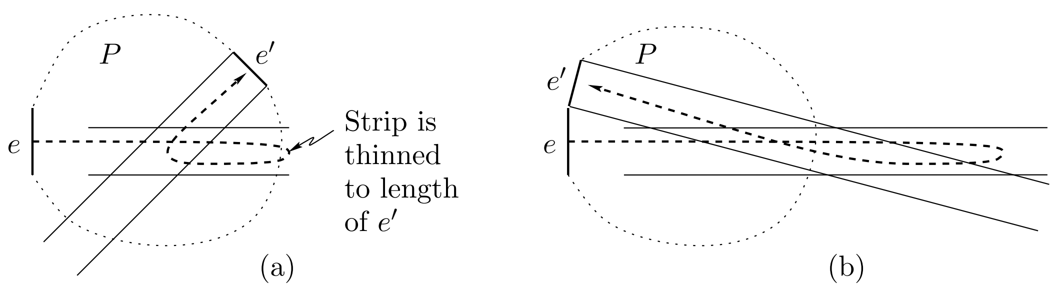

Fig. 18. Covering a polygon P entering from edge e, and turning to the next edge e'. (a) e' has negative slope. (b) e' has positive slope.

Once we reach the edge e', we immediately reorient so that e' is vertical. We apply Theorem 11 to resize the strip along e' to the vertical extent of P'. Finally, if P and P' have opposite colors, we reverse the strip color along e' by applying the gadget in Fig. 6.

Folding the excess paper underneath completes the convex-decomposition method, thereby proving Theorem 10: any polyhedron can be wrapped with seams precisely along the edges of a convex decomposition. This also provides our third proof of Theorem 1, since the time bound is clearly polynomial in n and the bound on the number of folds follows from the use of $O(n)$ hide gadgets.

6 Conclusion

We have described three methods for constructing an arbitrary silhouette, or more generally a 2-colored polyhedron. The first method uses triangles as building blocks, following a Hamiltonian triangulation refinement of the polyhedron. The second method uses "rings" as building blocks, following a natural ring decomposition from the straight skeleton. We note that these two approaches resemble the two main algorithms for "milling" a pocket: zig-zag and contour machining [5]. The third method uses seamless convex polygons as building blocks, allowing us to control the pattern of seams in the overall folding (e.g., to optimize the number or length of seams) subject to convexity of the seamless regions.

Acknowledgements

We thank Jin Akiyama for introducing us to the "gift wrapping" problem. We thank Anna Lubiw and Ian Munro for valuable comments on this work. The first author is partially supported by NSERC. The third author is partially supported by NSF grants CCR-9504192 and CCR-9732220, and by grants from Boeing, Bridgeport Machines, Hughes Research Labs, Sandia, and Sun Microsystems.

References

[1] Oswin Aichholzer and Franz Aurenhammer, Straight skeletons for general polygonal figures in the plane, in: Proceedings of the 2nd Annual International Computing and Combinatorics Conference, Lecture Notes in Computer Science, Vol. 1090 (Hong Kong, 1996) 117–126.

[2] Oswin Aichholzer, Franz Aurenhammer, David Alberts, and Bernd Gärtner, A novel type of skeleton for polygons, Journal of Universal Computer Science 1 (1995) 752–761.

[3] Jin Akiyama, Why Taro can do geometry, in: Proceedings of the 9th Canadian Conference on Computational Geometry (1997) 112. Invited talk.

[4] Jin Akiyama, Takemasa Ooya, and Yuko Segawa, Wrapping a cube, Teaching Mathematics and Its Applications 16 (1997) 95–100.

[5] Esther M. Arkin, Sándor P. Fekete, and Joseph S. B. Mitchell. Approximation algorithms for lawn mowing and milling. Technical Report 97.255, Angewandte Mathematik und Informatik, Universität zu Köln, 1997. Submitted to *Computational Geometry: Theory and Applications*.

[6] Esther M. Arkin, Martin Held, Joseph S. B. Mitchell, and Steven S. Skiena, Hamiltonian triangulations for fast rendering, The Visual Computer 12 (1996) 429–444.

[7] Marshall Bern and Barry Hayes, The complexity of flat origami, in: Proceedings of the 7th Annual ACM-SIAM Symposium on Discrete Algorithms (Atlanta, 1996) 175–183.

[8] Bernard Chazelle and D. P. Dobkin, Optimal convex decompositions, in: G. T. Toussaint, ed., *Computational Geometry* (North-Holland, Amsterdam, 1985) 63–133.

[9] David Eppstein and Jeff Erickson, Raising roofs, crashing cycles, and playing pool: Applications of a data structure for finding pairwise interactions, in: Proceedings of the 14th Annual ACM Symposium on Computational Geometry (Minneapolis, Minnesota, 1998) 58–67.

[10] D. H. Greene, The decomposition of polygons into convex parts, in: F. P. Preparata, ed., *Computational Geometry* (JAI Press, London, 1983) 235–259.

[11] Taichiro Hasegawa, *Magical Origami: Alphabets and Numerals* (Seibundoh Shinkosha, 1996).

[12] Kunihiko Kasahara, *Origami Omnibus* (Japan Publications, Inc., 1988).

[13] Kunihiko Kasahara and Toshie Takahama, *Origami for the Connoisseur* (Japan Publications, Inc., 1987).

[14] J. Mark Keil, Polygon decomposition, in: Jörg-Rüdiger Sack and Jorge Urrutia, eds., *Handbook of Computational Geometry* (Elsevier Science Publishers B.V. North-Holland, Amsterdam). To appear.

[15] J. Mark Keil, *Decomposing polygons into simpler components*, Ph.D. Thesis, University of Toronto, Toronto, Canada, 1983.

[16] J. Mark Keil, Decomposing a polygon into simpler components, SIAM Journal on Computing 14 (1985) 799–817.

[17] Robert J. Lang, A computational algorithm for origami design, in: Proceedings of the 12th Annual ACM Symposium on Computational Geometry (Philadelphia, PA, 1996) 98–105.

[18] Andrzej Lingas, The power of non-rectilinear holes, in: Proceedings of the 9th International Colloquium on Automata, Languages, and Programming, Lecture Notes in Computer Science, Vol. 140 (Springer-Verlag, Aarhus, Denmark, 1872) 369–383.

[19] E. A. Lord and C. B. Wilson, *The Mathematical Description of Shape and Form* (Ellis Horwood Limited, West Sussex, England, 1984).

[20] John Montroll, *African Animals in Origami* (Dover Publications, 1991).

[21] John Montroll, *Origami Inside-Out* (Dover Publications, 1993).