Folding and One Straight Cut Suffice*

Erik D. Demaine† Martin L. Demaine† Anna Lubiw†

Take a sheet of paper, fold it into some flat origami, and make one complete straight cut. What shapes can the unfolded pieces make? For example, Figure 1 shows how to cut out a five-pointed star in this way. You could imagine cutting out the silhouette of your favorite animal, object, or geometric shape.

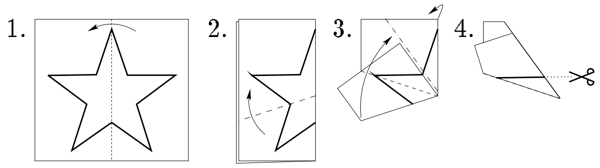

Figure 1: *How to fold and cut a five-pointed star.*

The first published reference to this *fold-and-cut* idea is an 1873 article in *Harper's New Monthly Magazine* [1]. This article tells the story of Betsy Ross showing George Washington how easily a five-pointed star could be made for the American flag, by folding a sheet of paper and making one straight cut with scissors.

Folding and cutting has also been used for a magic trick by Houdini, before he became a famous escape artist [8]. Another magician, Gerald Loe, studied this idea in some detail; his *Paper Capers* [11] describes how to cut out arrangements of various geometric objects, such as a circular chain of stars. Martin Gardner wrote about this problem in his famous series in *Scientific American* [7]. He was particularly impressed with Loe's ability to cut out any desired letter of the alphabet.

Gardner [7] was the first to state cutting out complex polygons as an open problem. What are the limits of this fold-and-cut process? What polygonal shapes can be cut out?

In the full version of this paper [4], we prove

Theorem 1. *Given any collection of straight edges, there exists a flat folding and a line in that folding such that cutting along it results in the desired pattern of cuts.*

This includes multiple disjoint, nested, and/or adjoining polygons, as well as floating line segments and points: a general plane graph. Here a *plane graph* is a graph with a fixed planar straight-line embedding. To solve this problem, we present an algorithm that computes the creases and the actual flat origami that lines up precisely the given plane graph. Cutting along this line hence achieves the desired result.

Some examples of the crease patterns resulting from our algorithm are given in Figures 2 and 3.

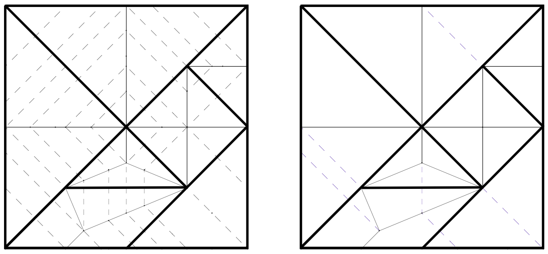

Figure 2: *Crease pattern for a Tangram set. The left shows all the perpendiculars, and the right shows a minimal set required for an actual folding.*

Inspired by preliminary versions of this work, Bern, Demaine, Eppstein, and Hayes [3] have proposed an alternative solution to the fold-and-cut problem using the idea of disk packing. This solution is more "local" than the one presented here, which exploits and demonstrates the global structure of the problem. The advantage of the disk-packing solution is that the number of folds is bounded in terms of the number of vertices and minimum feature size. On the other hand, the origamis presented here are more natural, often much simpler, and easier to fold in practice. Our techniques have also helped extend work in algorithmic origami design [9, 10].

A longer version of this paper appears in [5]. Here we describe the main creases for our solution, leaving the more difficult part of describing the final folded state for the full paper [4].

We refer to the plane graph that we want to line up as the *cut graph*, and to its vertices, edges, and faces as cut vertices, cut edges, and cut faces. The

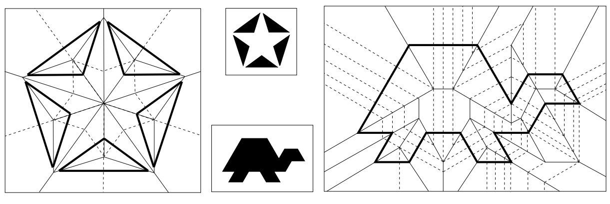

Figure 3: *Crease pattern for a fancy star and turtle.*

*Supported by NSERC.

†Dept. of Computer Science, University of Waterloo, Waterloo, Ontario N2L 3G1, Canada, email: eddemaine@uwaterloo.ca.

plane graph of creases is called the *crease pattern*, and consists of two main components, the straight skeleton and perpendicular folds. In the figures, we draw the cut graph with thick lines, the straight skeleton with thin lines, and the perpendicular folds with dashed lines.

A natural way to line up two cut edges is to fold along the bisector of their extensions. A generalization of this to arbitrary cut graphs is the *straight skeleton*. This structure is defined to be the trajectories of the vertices as we *shrink* the faces of the cut graph. This consists of insetting each cut vertex so that every shrunken cut edge is parallel to the original, and the distance between the shrunken and original cut edges is the same over all cut edges (at a particular time). Whenever a cut face becomes nonsimple, we recursively shrink the subregions. See Figure 4(a).

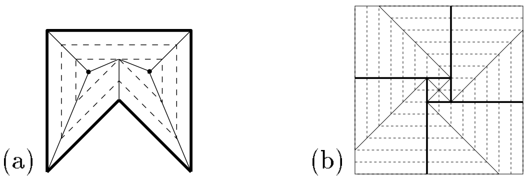

Figure 4: *(a) Shrinking a face of the cut graph to form the straight skeleton. (b) An example of spiraling.*

The straight skeleton was first defined for general plane graphs by Aichholzer and Aurenhammer [2], who presented an $O(n^2 \log n)$-time algorithm. Recently, Eppstein and Erickson [6] developed an $O(n^{17/11+\epsilon})$-time algorithm.

The straight skeleton by itself is not foldable. We can add a fold that is perpendicular to a cut edge, and maintain the property that the cut edges line up. More specifically, for each vertex of the straight skeleton, we add a collection of folds called a *perpendicular*. For a general point p, we recursively define the perpendicular associated with p as follows. For each (closed) skeleton face f that p is in, let l be the line going through p and perpendicular to (the line extending) the cut edge contained in face f. Let m be the connected piece of $l \cap f$ that touches p; this may be just p itself, a line segment of positive length, a ray, or a line. Then the perpendicular associated with p contains both m and the perpendiculars associated with the endpoints of m.

One interesting phenomenon that can happen with perpendiculars is *spiraling*. A simple example is shown in Figure 4(b), where the cut graph is a "pinwheel." The number of edges in each perpendicular depends on the size of the paper.

Unfortunately, this means that the number of creases is unbounded in terms of the number n of vertices, minimum feature size, or similar metric. However, all the edges of a perpendicular fold to a common line. So in fact the more natural combinatorial object is a set of points folding to become collinear. While each perpendicular bends an unbounded number of times, there are only $O(n)$ perpendiculars [4].

To proceed beyond the folding of perpendiculars, we consider the faces in the plane graph of perpendiculars, called *corridors*. Each corridor is bounded by two perpendiculars that are a constant width apart, and folds up like an accordion. Corridors can have two topologies: most are *linear*, having one "end" on each side, and the rest are *circular*, such as the one in the middle of Figure 2 (left).

While the crease pattern presented here is simple and intuitive, the description of the resulting flat origami is difficult. This consists of specifying the positions and overlap order of crease-pattern faces in the final folded state; in particular, we must specify which folds are mountains and which are valleys. Constructing a folded state, and showing certain properties such as noncrossing (that the paper does not self-intersect), are required for a proof of correctness for the algorithm. We refer the reader to [4] for details.

Acknowledgments. We thank Marshall Bern, Greg Frederickson, Tom Hull, and Rob Lang for valuable discussions.

References

[1] National standards and emblems. *Harper's New Monthly Magazine*, 47(278):171–181, 1873.

[2] O. Aichholzer and F. Aurenhammer. Straight skeletons for general polygonal figures in the plane. In *Proc. Int. Computing and Combinatorics Conf.* (COCOON), Hong Kong, pp. 117–126, 1996.

[3] M. Bern, E. Demaine, D. Eppstein, and B. Hayes. A disk-packing algorithm for an origami magic trick. In *Proc. Int. Conf. Fun with Algorithms*, Elba, 1998.

[4] E. D. Demaine, M. L. Demaine, and A. Lubiw. Folding and one straight cut suffice. Technical Report CS-98-18, University of Waterloo, 1998.

[5] E. D. Demaine, M. L. Demaine, and A. Lubiw. Folding and cutting paper. In *Proc. Japan Conf. Discrete and Computational Geometry*, Tokyo, 1998.

[6] D. Eppstein and J. Erickson. Raising roofs, crashing cycles, and playing pool: Applications of a data structure for finding pairwise interactions. In *Proc. Symp. Computational Geometry*, Minn., 1998, pp. 58–67.

[7] M. Gardner. Paper cutting. In *New Mathematical Diversions (Revised Edition)*, pp. 58–69. MAA, 1995.

[8] H. Houdini. *Paper Magic*, pp. 176–177. E. P. Dutton & Company, 1922.

[9] R. J. Lang. A computational algorithm for origami design. In *Proc. Symp. Computational Geometry*, Philadelphia, 1996, pp. 98–105.

[10] R. J. Lang. *TreeMaker 4.0: A Program for Origami Design* (manual), 1998.

[11] G. M. Loe. *Paper Capers*. Magic, Inc., 1955.