The Complexity of Flat Origami

Marshall Bern *   Barry Hayes †

(Extended Abstract)

Abstract

We study a basic problem in mathematical origami: determine if a given crease pattern can be folded to a flat origami. We show that assigning mountain and valley folds is NP-complete. We also show that determining a suitable overlap order for flaps is NP-complete, even assuming a valid mountain and valley assignment.

1 Introduction

Origami is the centuries-old art of folding paper into sculpture. In this article, we investigate the problem of determining if a sheet of paper, with a given pattern of lines representing the locations of folds, can be folded flat. This question is fundamental to any mathematical theory of origami and hence has already attracted some attention. The question is also fairly natural from a practical point of view. Assuming the input to be a crease pattern (rather than, say, a target three-dimensional solid) is more reasonable than it may first appear, since new origamis ("models") are typically created by modifying known crease patterns. Limiting attention to flat origami is justified by the fact that most models fold flat up until the finishing steps and some, including the traditional crane, fold flat even after completion.

Previous Work. A number of contemporary origami artists, notably Engel, Fuse, Lang, and Maekawa, apply geometric heuristics to origami design. One heuristic uses the centers of non-overlapping disks to determine the tips of "flaps" [9]. Another technique [1] builds complicated crease patterns out of repeating blocks called "molecules".

There is also a smattering of explicitly mathematical work on origami [4, 5, 6, 8, 9], almost all of it concerned with flat foldability. Our paper was inspired by, and answers some of the questions implicit in, a recent paper by Hull [4].

New Results. We consider three versions of the problem of flat foldability. If we only require that the origami fold flat within a neighborhood of each vertex, then the problem is easy. For this (somewhat artificial) problem, we give a linear-time algorithm that assigns "mountain" and "valley" orientations and determines an overlap order for flaps. However, if we require that the origami fold flat everywhere, then it is NP-complete to find appropriate orientations and overlap order. Finally, if orientations are given, just finding an overlap order is NP-complete. Together our results show that the real difficulty of the problem does not lie in simultaneously handling all vertices, but rather in avoiding edge-edge collisions.

*Xerox PARC, 3333 Coyote Hill Rd., Palo Alto, CA 94304, bern@parc.xerox.com

†ParcPlace Systems, Inc., 999 E. Arques Ave., Sunnyvale, CA 94086, bhayes@strider.stanford.edu

2 Definitions

Our definitions vary a bit from those of other authors [4, 8] in order to avoid some technical difficulties. We omit or sketch proofs in this extended abstract.

Definition 1. *A* **crease pattern** *is a finite planar straight-line graph drawn on a convex planar region (the* **paper***). A* **crease** *is an edge of the planar graph.*

Unless we state otherwise, we shall assume that the paper is the unit square. The operative part of the next definition is the requirement that an origami be a one-to-one mapping, modeling the fact that (physical) paper cannot penetrate itself.

Definition 2. *An* **origami** *is a continuous, one-to-one mapping of a crease pattern to* $\mathbb{R}^3$*. The mapping must be smooth (differentiable) everywhere except along creases.*

The mapping need not be differentiable at creases, but along each crease the dihedral angle (defined locally) must be a smooth function. It is convenient to measure dihedral angles by deviation from flatness, so that sharp folds are close to π or $-\pi$ radians.

Definition 3. *A* **flat origami** *is an infinite sequence of origamis with the same crease pattern, such that the images of each crease converge to a line segment with dihedral measuring either* π *or* $-\pi$ *and the images of each face converge to a planar polygonal region, congruent to the face. Convergence is not just pointwise, but sufficiently strong that metric properties converge as well.*

We shall intuitively talk about flat origamis as single embeddings rather than sequences of embeddings, with the understanding that the statement holds in the limit. We shall call a crease pattern *flat foldable* if it is the crease pattern of some flat origami. A crease of a flat origami is called a *mountain* if its limiting dihedral is $-\pi$ and a *valley* if its limiting dihedral is π. Intuitively, a mountain points up and a valley points down if the paper is unfolded back to a square.

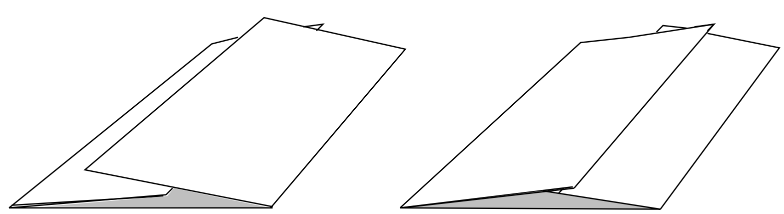

Figure 1. Two origamis with the same MV-assignment but different overlap orders.

A flat origami can be described by its *MV-assignment*, a mapping from the set of creases to the set $\{-\pi, \pi\}$. This description, however, is not complete, as origamis with the same MV-assignment may overlap differently, as shown in Figure 1. To give a complete description, we orient the flat origami so that all its faces are parallel to the xy-plane and then project all creases, along with the boundary of the square, down onto the plane. This forms a cell complex with $O(n^2)$ cells, where n is the number of creases. The *overlap map* gives the vertical order of origami faces above each cell of this arrangement.

The overlap map can be encoded with $O(n^2)$ bits by recording for each pair of origami faces, which face lies on top; the vertical relationship between pairs remains constant throughout the origami. This encoding may be quite inefficient, because we know of no crease pattern with more than $2^{O(n \log n)}$ different flat origamis. Figure 2(a) shows such a crease pattern. A few initial folds make the square paper into a rectangle, which is then folded into a U-shape (shown opened up a bit for clarity). The lower arm is pleated horizontally and the upper vertically, so that each of the $\Omega(n)$ tabs on the lower arm can be tucked into any of the $\Omega(n)$ slots on the upper. Thus this crease pattern has $n^{\Omega(n)} = 2^{\Omega(n \log n)}$ different flat origamis. For rectangular paper with aspect ratio about n, there are crease patterns with $2^{\Omega(n^2)}$ different flat origamis, as shown in Figure 2(b).

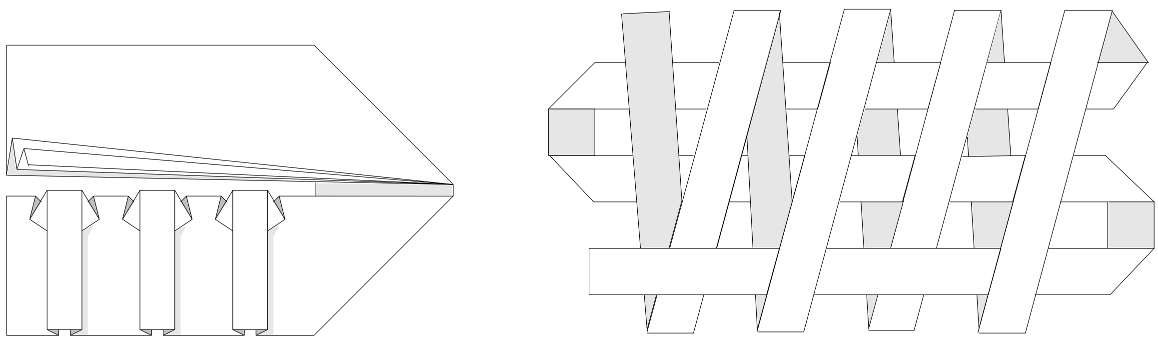

Figure 2. (a) There are $2^{\Omega(n \log n)}$ ways to tuck the tabs into the accordion. (b) There are $2^{\Omega(n^2)}$ different ways to weave the horizontal sections between the diagonals.

3 Single Vertex Flat Folding

We first review some well-known necessary conditions for crease patterns to fold flat in the neighborhood of a single vertex. Let v be an interior vertex of crease pattern $\mathcal{C}$, and assume that there is a flat origami with crease pattern $\mathcal{C}$. Center a small sphere at v, such that the intersection of the flat origami and the sphere is a flat spherical polygon P_v, as in Figure 3. Convex and reflex vertices of P_v correspond to mountains and valleys (shown as dotted and dashed lines, respectively), and the arc lengths of P_v are proportional to the angles between creases meeting at v. The requirement that the flat polygon closes up has the following consequences (see [4, 7, 8]).

(K1) (Kawasaki) The sum of alternate angles around v is π.

(M) (Maekawa) The number of mountain folds minus the number of valley folds meeting at v is either 2 or -2.

Kawasaki's condition implies that a flat-foldable crease pattern is a convex subdivision. Maekawa's condition implies that v must be incident to an even number of creases. Now let the creases around v be $e_1, e_2, \ldots, e_k$, with $e_1 = e_{k+1}$. Let the angle between e_i and e_{i+1} measure α_i. The requirement that the polygon not cross itself has the following consequence.

(K2) (Kawasaki) If $\alpha_i < \alpha_{i-1}$ and $\alpha_i < \alpha_{i+1}$, then e_i and e_{i+1} must have opposite assignments.

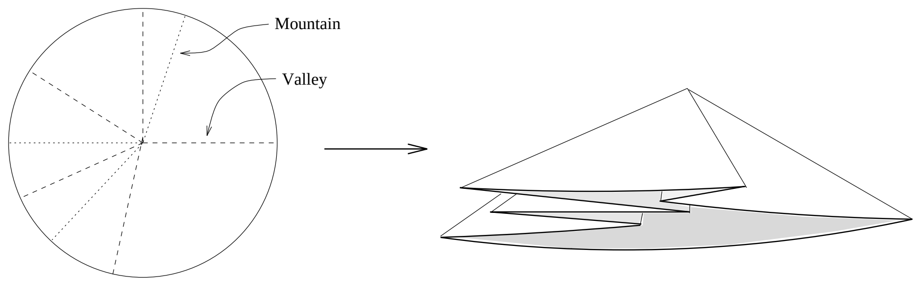

Figure 3. A circle around an interior vertex in a crease pattern folds to a flat spherical polygon.

If v is a boundary vertex of the crease pattern, lying along an edge or a corner of the paper, P_v will be an open polygonal chain, and hence only condition (K2) applies. For boundary vertices we define α_1 and α_k to be the angles between the first and last creases and the edge of the paper; we do not consider these angles to be adjacent.

We now consider the converse: sufficient conditions. Let $\mathcal{D}$ be a disk with a planar straight-line graph drawn on it. Assume that $\mathcal{D}$ contains only one vertex v, which lies at the center of the disk, along with some number of creases radiating out from v. Kawasaki [8] proved that (K1) is a sufficient condition for $\mathcal{D}$ to have a mountain and valley assignment that makes it flat foldable. We give our own proof, an algorithm that will find use in Section 4.

Single-Vertex Algorithm. Our algorithm uses condition (K2) recursively. Again let $e_1, e_2, \ldots, e_k, e_{k+1} = e_1$ denote the creases in order around v, and let α_i denote the angle between e_i and e_{i+1}.

The algorithm finds a locally minimal angle α_m, that is, an m such that $\alpha_m \leq \alpha_{m+1}$ and $\alpha_m \leq \alpha_{m-1}$. The algorithm then subtracts α_m from α_{m+1}, removes e_m and e_{m+1}, and merges α_{m-1} and the remainder of α_{m+1} into one angle. These steps cut a pie-shaped wedge from $\mathcal{D}$ and produce a new crease pattern $\mathcal{D}'$. The algorithm then calls itself recursively to compute a flat origami for $\mathcal{D}'$. (Here we have slightly generalized the notion of crease pattern, since the angles in $\mathcal{D}'$ do not sum to 2π.) The removed wedge can be attached to the flat origami for $\mathcal{D}'$ in either one of two ways, corresponding to the two possible assignments for e_m and e_{m+1}: M and V or V and M. See Figure 4. In either case, the wedge should be next to the side to which it is attached in the vertical order, or else the flat polygon may self-intersect.

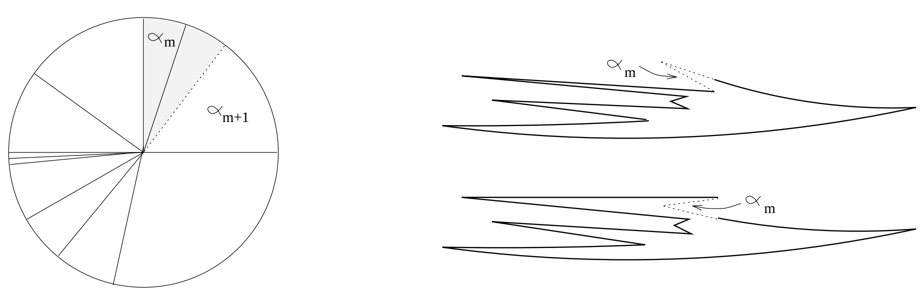

Figure 4. A recursive algorithm for folding a single-vertex flat origami.

The recursion bottoms out when there are only two creases left; (K1) implies that the last two angles must be equal. The algorithm assigns the creases to be both M or both V;

this free choice determines which side of the spherical polygon P_v is the interior.

The algorithm also solves the easier problem of flat folding an arbitrary boundary vertex (without any condition on angles). In this case, the recursion can bottom out with zero, one, or two creases left. The last assignment is arbitrary, unless there are two creases left and the minimum angle is between the two creases.

Notice that in this algorithm, each crease e_i is paired with a *partner* e_j, such that e_i and e_j have opposite assignments or—in the case of the last pair—the same assignments. Other choices—which of e_i and e_j is M, and which locally minimal angle to process next—are unconstrained.

It is not hard to implement this algorithm so that it runs in linear time. An initial linear-time pass finds all locally minimal angles and places them on a queue. After removing the first angle and merging the two neighboring angles, we update the queue by removing angles that are no longer locally minimal and adding angles that have become locally minimal—$O(1)$ operations in all.

4 All Vertices Simultaneously

We now have a condition—that alternate angles sum to π—which determines whether there exists an origami that is flat in the neighborhood of a single vertex. For the remainder of this paper, we shall assume that all crease patterns satisfy this angle condition at each interior vertex.[1] In this section, we take a halfway step towards the problem of deciding the global flat foldability of a crease pattern.

Definition 4. *A* **vertex-flat origami** *is an infinite sequence of origamis with the same crease pattern, such that around each vertex there is a ball within which faces converge to planar pie-shaped wedges and edges converge to line segments with dihedrals measuring either π or $-\pi$ (with both ends of an edge in agreement).*

A vertex-flat origami is not the same as a flat origami, because it may be quite far from flat away from vertices. In particular, edges and faces may bend and stretch to avoid collisions.

Definition 5. VERTEX FLAT FOLDABILITY *is the problem of determining whether or not a given n-vertex crease pattern is the crease pattern of a vertex-flat origami.*

We require only a yes/no answer for VERTEX FLAT FOLDABILITY, in part because we have not defined a representation such as the overlap map for arbitrary origamis. We now give a polynomial-time algorithm for VERTEX FLAT FOLDABILITY. For yes instances of the problem, this algorithm finds an MV-assignment that satisfies conditions (M) and (K2) (and also (K2) applied iteratively as in the single-vertex algorithm). Incidentally, (M) alone gives the following interesting matching problem: assign M and V to edges such that at each vertex the number of M edges minus the number of V edges is plus or minus two. This problem is polynomially solvable even for general graphs (see [10], page 389).

Theorem 1. *There is a linear-time algorithm for* VERTEX FLAT FOLDABILITY.

[1] Checking angle condition (K1) is trivial if we allow exact real-number arithmetic, but it is unclear how to check (K1) assuming only finite-precision arithmetic and rational coordinates for vertices of $\mathcal{C}$.

Proof: We start by imposing a general position condition that we shall later remove: at each vertex v in crease pattern $\mathcal{C}$, at each step of the single-vertex algorithm given above, no angle has the same measure as one adjacent to it.

We use the pairing of creases in the single-vertex algorithm to group creases of $\mathcal{C}$. If e_i is the partner of e_j at vertex v, then we place e_i and e_j into the same group. (This grouping is similar to the "origami line graph" of Hull [4] and Justin [6].) Since a crease has exactly one partner at an interior vertex (and at most one partner at a boundary vertex), each group forms either a cycle or a path. Assigning M or V to any single crease within a group sets all the creases. A cycle (but never a path) may be self-contradictory, or *inconsistent*, for example, an odd cycle in which each partnership pair must be oppositely assigned. Figure 5 shows an example [4] of a crease pattern with an inconsistent cycle. It is clear that such a crease pattern is not vertex-flat foldable; below we show that, conversely, a crease pattern with no inconsistent cycles is vertex-flat foldable.

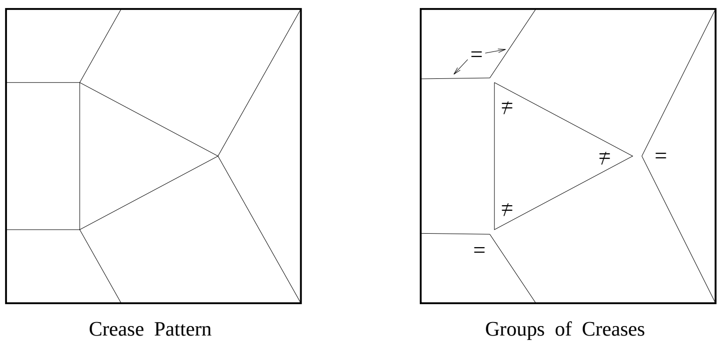

Figure 5. A crease pattern with an inconsistent cycle.

We now remove the general position condition. Assume that at each vertex, we run the single-vertex algorithm as before and obtain some grouping of creases into cycles and paths. Our plan is to merge two groups that meet at a tie. Assume the following: at some step of the single-vertex algorithm, crease e_i forms the same angle with e_{i-1} as it does with e_{i+1}; initially e_i was paired with e_{i+1} and e_{i-1} with e_j; and e_i and e_{i-1} belong to two different groups. If e_i had instead been paired with e_{i-1}, e_{i+1} could have been paired with e_j, because e_{i+1} forms the same angle with e_j that e_{i-1} did after subtracting α_i from α_{i+1}. By changing the pairing, we can merge e_i and e_{i-1}'s groups, as shown in Figure 6. This merging procedure can be applied repeatedly in the case of more than one tie.

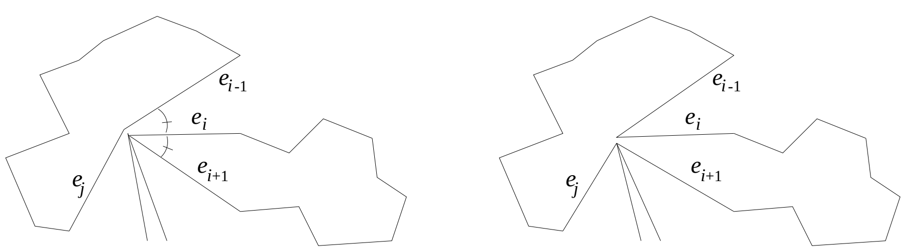

Figure 6. Groups formed by tie-breaking choices can be merged.

It is never disadvantageous to merge an inconsistent cycle with another group. If the other group is a path, the result of the merger is a larger path, hence consistent. If the other group is a consistent cycle, the result is a larger inconsistent cycle with at least as many opportunities for mergers. And if the other group is another inconsistent cycle, the inconsistencies eliminate each other and the result is a consistent cycle.

We can now fill in the complete algorithm. Run the linear-time single-vertex algorithm, breaking ties arbitrarily. Form groups of edges with a linear-time traversal of the graph. Now try to resolve inconsistent cycles with mergers. Each merger causes a "union" operation in a union-find data structure [11] representing the groups and changes $O(1)$ pointers in a planar graph data structure representing the crease pattern. Because unions only occur between edges adjacent around a vertex, the overall running time remains linear [2].

Crease pattern $\mathcal{C}$ has a vertex-flat origami if and only if all inconsistent cycles can be eliminated. One such vertex-flat origami leaves the paper unfolded except near the vertices, which are pulled above or below the plane of the paper. Within a small ball around each vertex, the paper is folded as in a single-vertex flat origami. Slightly farther away, faces curve to meet the main plane of the paper. The curving is such that the intersection of the paper with an expanding sphere around the vertex grows from a flat spherical polygon into the equator of the sphere, without ever self-intersecting (a homotopy of a polygon to a circle). ∎

5 Flat Foldability

This section and the next give NP-completeness reductions. In each case, we reduce the following NP-complete problem [3] to the origami problem.

Definition 6. NOT-ALL-EQUAL 3-SAT *is given by a collection of clauses, each containing exactly three literals. The problem is to determine whether or not there exists a truth assignment such that each clause has either one or two true literals.*

Our first target problem is FLAT FOLDABILITY, the canonical flat-origami problem.

Definition 7. FLAT FOLDABILITY *is the problem of determining whether or not a given crease pattern is the crease pattern of a flat origami.*

As usual in reductions to versions of SAT, we create constructions ("gadgets") for boolean variables and clauses which we interconnect by "wires". For us, a *wire* will be two closely-spaced parallel creases. The spacing is close enough that in any flat folding the two creases in a wire must have opposite assignments, forming a "pleat". In order to distinguish left from right, we shall label the wires in our gadgets with directions. These directions serve only as expository devices; they are not part of the crease pattern. We shall call a wire in an MV-assignment *true* (respectively, *false*) if the valley crease lies to the right (left) of the mountain crease when facing along the wire's direction.

Figure 7(a) shows a clause gadget. This crease pattern consists of three wires that meet at a central equilateral triangle. We think of all three wires as directed into the triangle. There are eight MV-assignments that satisfy conditions (M) and (K2), the four shown in Figure 7(b) and the four obtained from these by reversing M and V. However, the rightmost assignment is not flat foldable, because its three mountain edges collide at a point above—out of the plane of—the triangle. Of course, the same is true of its reversal.

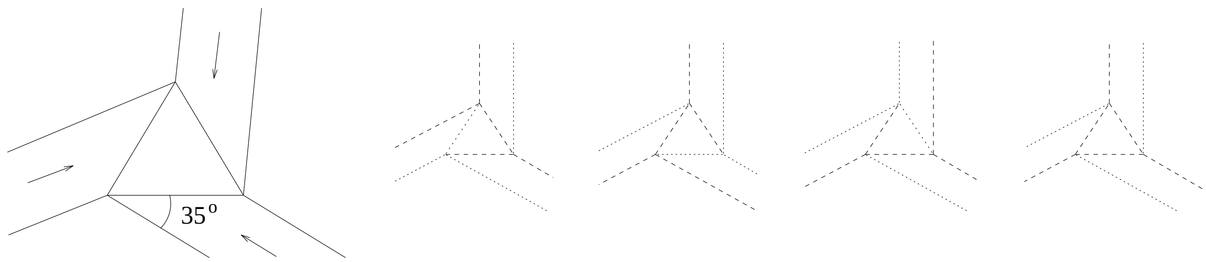

Figure 7. (a) The crease pattern of the clause gadget. (b) Possible assignments.

Lemma 1. *The clause crease pattern is flat foldable if and only if one or two of the incoming wires are true.*

We construct the truth-setting part of our reduction out of gadgets called *reflectors*. A reflector crease pattern consists of three wires that meet at an isosceles triangle with largest angle in the range $[90°, 180°)$, as shown in Figure 8(a). By varying the angle of the isosceles triangle, we obtain a one-parameter family of reflectors.

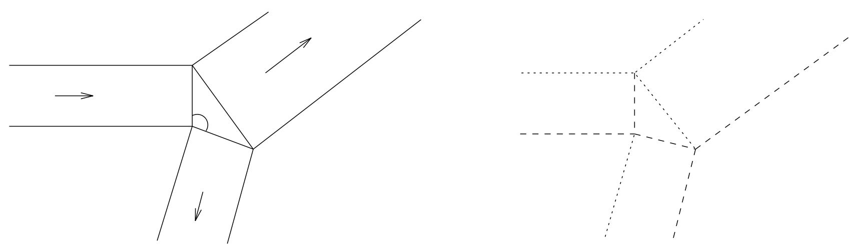

Figure 8. Reflector crease pattern and possible MV-assignment.

Lemma 2. *The reflector crease pattern is flat foldable if and only if the incoming wire agrees with the outgoing broad wire and disagrees with the outgoing narrow wire.*

The last gadget, called a *crossover* and shown in Figure 9, lets wires cross while preserving their truth settings. The angles at which the wires meet the central parallelogram are chosen so that each wire folds over or under the parallelogram's center, forcing continuity of truth settings. Our reduction uses crossovers of 90° and 135°, as shown.

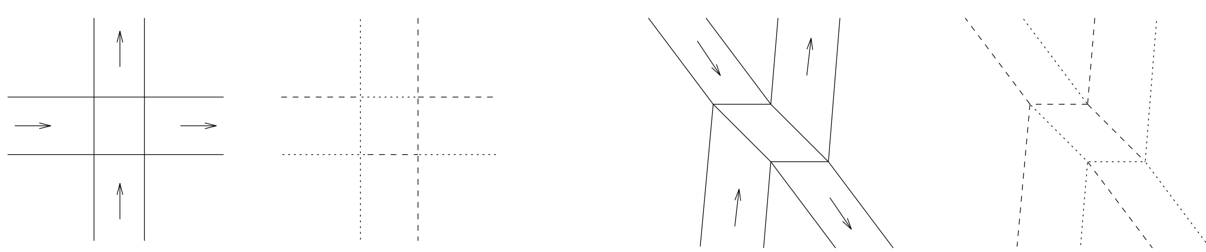

Figure 9. Crossovers of 90° and 135°.

Lemma 3. *The crossover crease patterns are flat foldable if and only if each opposite pair of incoming and outgoing wires agree.*

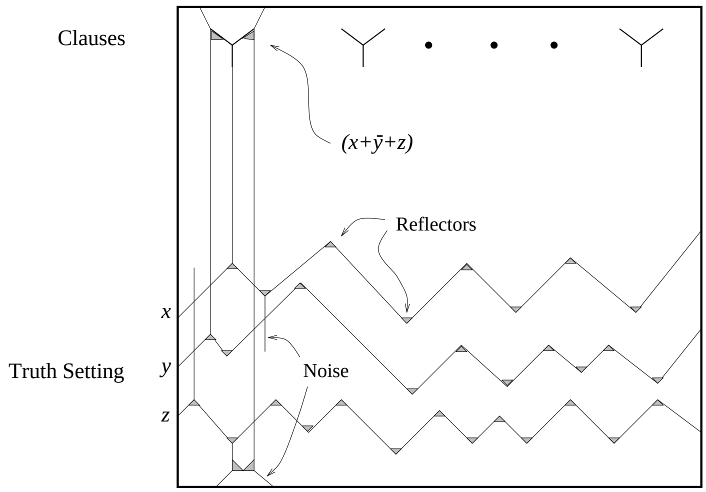

Figure 10. Schematic of the entire reduction.

We now put these gadgets together. Figure 10 shows a schematic of the entire reduction. Gadgets are spaced quite far apart relative to the width of wires, so that any flat origami with this crease pattern would look like a slightly wrinkled square sheet of paper. At the top of the paper, we place one clause gadget for each clause in the boolean formula. Each wire entering a clause gadget has width $\lambda > 0$.

At the bottom of the paper, we place one truth-setting construction for each boolean variable. A truth-setting construction includes a right-angled zig-zag formed by a wire of width $\lambda/\sqrt{2}$ with reflectors at each turn. We consider zig-zags to be directed from left to right. The orientation of the folds—zigs true and zags false or the other way around—determines whether the boolean variable is true or false. For each use of boolean variable x in a clause, we send a wire of width λ upwards from a reflector along the zig-zag for x. To obtain uncomplemented x, we can either take the signal from an upper reflector and use no additional reflectors on the path to the clause gadget or take the signal from a lower reflector and use three additional reflectors on the path to the clause. Similarly, there are two ways to obtain $\bar{x}$. Additional reflectors send out extraneous "noise" wires.

It is not hard to confirm that the crease pattern just described will be flat foldable exactly when the original boolean formula has an assignment that makes one or two literals true in each clause. A final point concerns numerical precision. As we have explained our gadgets, some vertices must have irrational coordinates. However, it is not too hard to show that the same construction can be carried out using rational coordinates with the total number of bits only polynomial in the size of the NOT-ALL-EQUAL 3-SAT instance.

Theorem 2. FLAT FOLDABILITY *is NP-complete.*

6 Assigned Flat Foldability

In this section we prove that the origami problem remains hard even if we know a valid MV-assignment. We again use NOT-ALL-EQUAL 3-SAT as our starting problem. This reduction is quite intricate, so we omit most of the details in this extended abstract.

Definition 8. Assigned Flat Foldability *is the problem of determining whether or not there exists a flat origami with a given crease pattern and MV-assignment.*

The overall strategy is similar to the previous reduction. A wire will now consist of four parallel creases, with M, V, V, and M orientations. A wire is *true* if the left pleat is folded over the right pleat and *false* if the right overlaps the left, again defining left and right relative to nominal directions. (Figure 1 shows something similar.) We make the distance between the two pleats—the middle channel of three parallel channels—slightly larger than the width of the pleats themselves.

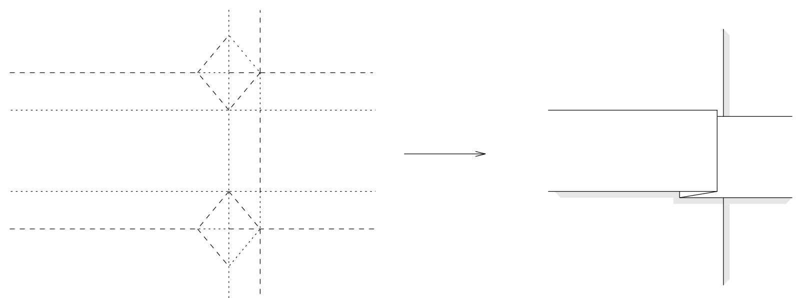

Figure 11. A tab is a rectangular flap folded into the paper in a preprocessing step.

This time our gadgets use a subgadget called a *tab*, shown in Figure 11. A tab is a rectangular flap folded into the paper before the creation of the gadgets; we use them to restrict the set of possible overlap maps. Tabs send out "noise" pleats that cross over signal wires. Crossovers, however, are easier than in the previous case: we can let a wire cross a pleat or another wire without any danger of the signal inverting. Thus we think of tabs as built into the paper, and we do not show the noise pleats in subsequent figures. We can make a tab arbitrarily long and thin—think of a diving board—by pleating the entire paper down the middle of the tab.

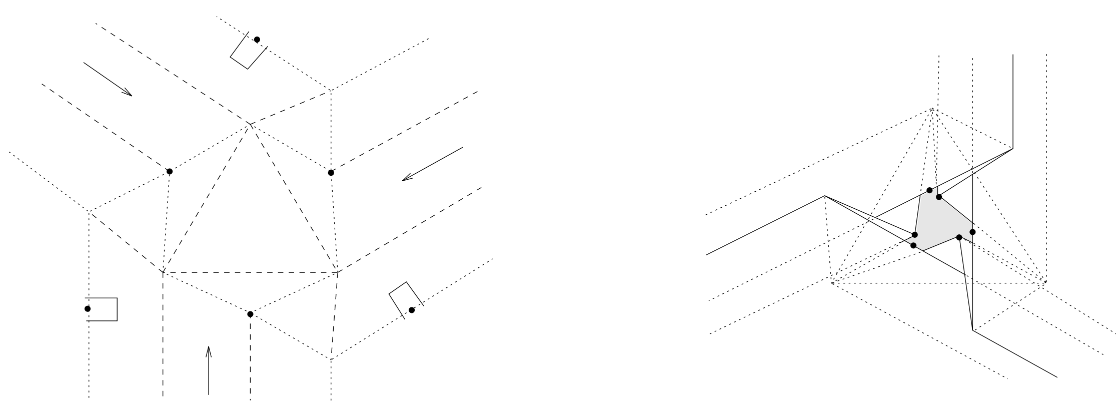

Figure 12. (a) Clause gadget for Assigned Flat Foldability. (b) Folded (tabs not shown).

The new clause gadget is shown in Figure 12(a). The points marked with dots fold to be very nearly coincident. They would be exactly coincident if the three widths within a wire were all the same; as is, they leave a small window at the center of the folded gadget, as shown in Figure 12(b). If we add tabs, represented by open boxes in Figure 12(a), to the

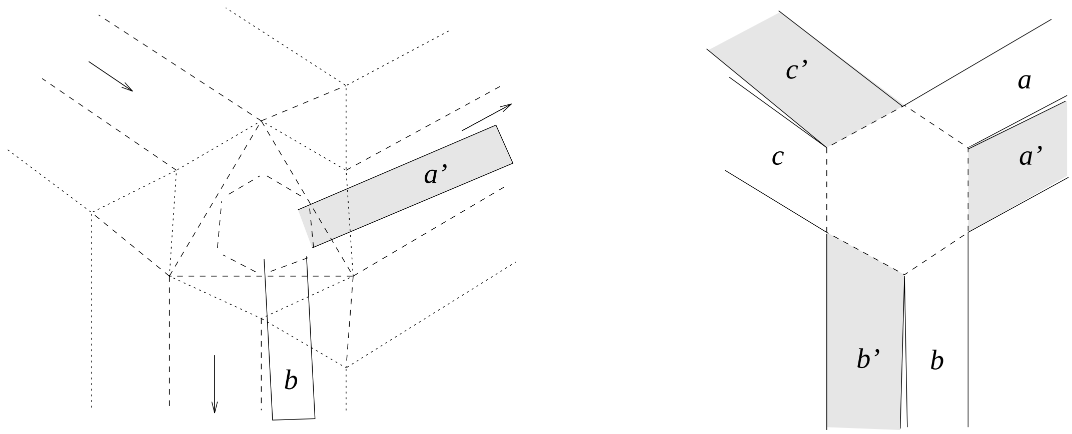

Figure 13. (a) Signal splitter. (b) Tabs form a hexagon with only two possible foldings.

sides of this window, we can eliminate the two "spiral" foldings, which correspond to the all-true and all-false settings.

Figure 13(a) shows a signal *splitter*, the gadget that takes the place of the reflector. The splitter starts with the same crease pattern and MV-assignment as the clause gadget, but uses tabs to eliminate all foldings *except* the two spirals, forcing the two output wires to disagree with the one input wire. The splitter gadget is too complicated to explain in detail here, but we can convey the general idea.

Imagine cutting a hexagon, with six arms attached, from a sheet of paper, as shown in Figure 13(b). For this cut-out to fold flat with each side of the hexagon forming a valley fold, shaded and unshaded sets of arms must each form spirals, for example, a above b above c above a. There are exactly two overlap maps for this flat origami: shaded spiral below unshaded spiral, and vice versa. The tabs in the splitter—only two of the six are shown—form a (slightly warped) set of hexagon arms. The truth setting of the input wire determines the overlap of tabs a' and b; the other tabs (not shown) then force the settings of the output wires.

The truth-setting part of the overall construction changes slightly from the previous reduction. Rather than take signals from the bottom of zig-zags and bounce them back upwards, we run a zig-zag for each of x and $\bar{x}$, with the two zig-zags linked by an inverter. (Alternatively, we could start from the problem NON-NEGATED NOT-ALL-EQUAL 3-SAT, in which each literal is uncomplemented.)

Theorem 3. ASSIGNED FLAT FOLDABILITY *is NP-complete.*

7 Conclusion

The mathematical study of origami is fairly new, so we close by suggesting some open problems for future research.

1. How many different flat origamis can there be with the same crease pattern? (This question is related to the solved problems of "stamp" and "map" folding.)

2. Does FLAT FOLDABILITY remain NP-complete for special inputs? Our reduction uses only degree-4 vertices, but perhaps some other restriction renders the problem

easy. (In particular, it is intriguing that our clause gadget resembles a well-known non-regular triangulation.)

3. Our use of tabs in the reduction for ASSIGNED FLAT FOLDABILITY is somewhat unesthetic. Are they necessary? If "tabs" can somehow be ruled out, does the problem become polynomially solvable?

4. Is every simple polygon, when scaled sufficiently small, the silhouette of a flat origami? How many creases are necessary to fold an n-vertex polygon? How thick (number of layers of paper) must the origami be? This last question is motivated by the fact that in practice it is very difficult to simultaneously fold a large number of layers.

5. Which simple polygons are silhouettes of flat origamis? (In this question, the polygon has a fixed size.)

6. K-LAYER FLAT FOLDABILITY is the problem of determining, for a given crease pattern, whether or not there exists a flat origami that is at most k layers thick. Our reduction above shows that this problem is NP-complete for $k \geq 7$. What about k from 2 to 6? Similar questions can be posed for ASSIGNED FLAT FOLDABILITY.

Acknowledgments

We would like to thank Brian Ewins and Tom Hull for illuminating e-mail.

References

[1] P. Engel. *Folding the Universe: Origami from Angelfish to Zen*. Vintage Books, 1989. Dover, 1994.

[2] H. Gabow and R.E. Tarjan. A linear-time algorithm for a special case of the disjoint set union. In *Proc. 15th Annual ACM Symp. on Theory of Computing*, 1983, 246–251.

[3] M.R. Garey and D.S. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness*. W.H. Freeman and Co., 1979.

[4] T. Hull. On the mathematics of flat origamis. *Congressus Numerantium* 100 (1994), 215–224.

[5] D.A. Huffman. Curvature and creases: a primer on paper. *IEEE Trans. on Computers* Volume C-25 (1976), 1010–1019.

[6] J. Justin. Abstract in *2nd Int. Meeting of Origami Science*, 1994, Otsu, Japan.

[7] K. Kasahara and T. Takahama. *Origami for the Connoisseur*. Japan Publications, Tokyo, 1988.

[8] T. Kawasaki. On the relation between mountain-creases and valley-creases of a flat origami. In *Origami Science and Technology*. H. Huzita, ed., 1989, 229–237.

[9] R. Lang. Mathematical algorithms for origami design. *Symmetry: Culture and Science* 5 (1994), 115–152.

[10] L. Lovász and M.D. Plummer. *Matching Theory*. Annals of Discrete Mathematics 29, North-Holland, 1986.

[11] R.E. Tarjan. *Data Structures and Network Algorithms*. SIAM, 1983.