Infinite All-Layers Simple Foldability

Hugo A. Akitaya[*]   Cordelia Avery[†]   Joseph Bergeron[†]   Erik D. Demaine[†]

Justin Kopinsky[†]   Jason S. Ku[†]

Abstract

We study the problem of deciding whether a crease pattern can be folded by simple folds (folding along one line at a time) under the *infinite all-layers* model introduced by [ADK17], in which each simple fold is defined by an infinite line and must fold all layers of paper that intersect this line. This model is motivated by folding in manufacturing such as sheet-metal bending. We improve on [ABD+04] by giving a deterministic $O(n)$-time algorithm to decide simple foldability of 1D crease patterns in the all-layers model. Then we extend this 1D result to 2D, showing that simple foldability in the infinite all-layers model can be decided in linear time for both unassigned and assigned axis-aligned orthogonal crease patterns on axis-aligned 2D orthogonal paper. On the other hand, we show that simple foldability is strongly NP-complete if a subset of the creases have a mountain–valley assignment, even for axis-aligned rectangular paper.

1 Introduction

A classic problem in computational origami (indeed, the topic of the very first paper in the field [BH96]) is *flat foldability*: given a crease pattern (planar straight-line graph) on a polygonal piece of paper P, can P be folded flat isometrically without self-intersection while creasing at all creases (edges) in the crease pattern? The decision problem is NP-hard [BH96], even if the paper P is an axis-aligned rectangle and the creases are at multiples of $45°$ [ACD+15]. But even when a crease pattern does fold flat, the motion to achieve that folding can be complicated [DM01], making the process impractical in some physical settings.

Motivated by practical folding processes in manufacturing such as sheet-metal bending, Arkin et al. [ABD+04] introduced the idea of *simple foldability*—flat foldability by a sequence of simple folds. Informally, a *simple fold* is defined by a line segment and rotates a portion of the paper around this line by $\pm180°$, while avoiding self-intersection; refer to Section 2 for formal definitions. The problem also makes sense with 1D paper, where P is a line segment and creases are defined by points in P. Arkin et al. defined several models of simple folds; in particular, the *all-layers* model requires that a simple fold through one crease also folds through all layers overlapping that crease. For many models, they showed that simple foldability is polynomial for 1D paper, polynomial for rectangular 2D paper with axis-aligned creases, weakly NP-complete for rectangular paper with creases at multiples of $45°$, and weakly NP-complete for orthogonal paper with axis-aligned creases.

---

[*]Tufts University, hugo.alves_akitaya@tufts.edu

[†]CSAIL, Massachusetts Institute of Technology, {cavery,jbergero,edemaine,jkopin,jasonku}@mit.edu

creases	paper	assigned	unassigned	partial
1D	rectangle	$O(n)$	$O(n)$	Open
2D orthogonal	rectangle	$O(n)$	$O(n)$	NP-complete
2D orthogonal	orthogonal	Open	$O(n)$	NP-complete

Figure 1: Complexity of deciding whether a crease pattern is foldable in the infinite all-layers simple folding model: either with 1D creases on rectangular paper, 2D orthogonal creases on rectangular paper, or 2D orthogonal creases on orthogonal paper; and either fully-assigned, fully-unassigned, or partially-assigned creases.

1D Simple Folding: Arkin et al. [ABD+04] developed an algorithm to determine simple foldability of a 1D mountain–valley pattern in $O(n \log n)$ deterministic time and $O(n)$ randomized time in the all-layers model. In Section 3, we improve on these results by giving a deterministic $O(n)$-time algorithm to decide simple foldability of 1D crease patterns in the all-layers model. This result removes the logarithmic factor from the best previous deterministic algorithm, or equivalently, removes the randomization from the best previous $O(n)$ algorithm. Combining with other techniques from [ABD+04], this linear-time algorithm extends to 2D axis-aligned orthogonal crease patterns on axis-aligned rectangular paper.

2D Simple Folding: In a recent followup to Arkin et al. [ABD+04], Akitaya et al. [ADK17] extended the list of simple folding models, and for many models, showed *strong* NP-hardness for 2D paper. In particular, they introduced the *infinite all-layers* model of simple folds, studied here, which requires that each simple fold is defined by an infinite line, and that all layers of paper intersecting this line must be folded. This model is probably the most practical of all simple folding models; for example, Balkcom's robotic folding system [BM08] is restricted to this model. For an axis-aligned rectangle paper with axis-aligned creases (and for 1D paper), infinite and non-infinite simple fold models are equivalent [ADK17]. In this paper, we study the complexity of one of the few remaining open problems in this area [ADK17]: infinite all-layers simple foldability on axis-aligned orthogonal paper with axis-aligned creases (henceforth referred to as *orthogonal crease patterns*). The techniques for proving strong NP-hardness in [ADK17] did not seem to extend to this case. In Section 4, we prove that indeed, when the creases are *unassigned* (can freely fold mountain or valley), this problem can be solved in polynomial (indeed, linear) time, by using the 1D result from Section 3. On the other hand, we prove in Section 5 that, when the creases are *partially assigned* (some creases must fold mountain, some creases must fold valley, while others can freely fold mountain or valley), the problem becomes strongly NP-complete, even for an axis-aligned rectangle of paper (and thus also for the regular all-layers simple fold model [ADK17]). By contrast, the same problem (all-layers simple foldability of rectangular paper with axis-aligned creases) with either fully assigned or completely unassigned crease patterns can be solved in polynomial time [ABD+04]. Remaining open problems are summarized in Section 6.

2   Definitions

We base our definitions on those from [ADK17], specializing to the infinite all-layers model. Define the *folding plane* $\mathbb{P}$ be a copy of $\mathbb{R}^2$, and define a *piece of paper* P to be a connected polygon in $\mathbb{P}$ possibly with holes. A *crease pattern* is defined by (P, Σ) where Σ is the edge set of a planar straight-line graph embedded in P. Members of Σ are interior disjoint line segments contained

in P called *creases*. A *facet* is a connected open set in $P \setminus \Sigma$. A crease pattern (P, Σ) can have an *assignment* $\alpha : \Sigma \to \{M, V\}$, where $M = 1$ (mountain) and $V = -1$ (valley). A function $\alpha : \Sigma^- \to \{M, V\}, \Sigma^- \subset \Sigma$ is called a *partial assignment* of (P, Σ).

We define an *infinite all-layers simple fold* to be the operation that takes a crease pattern (P, Σ), possibly with a (partial) assignment, and a directed infinite line ℓ in $\mathbb{P}$, and returns a simpler crease pattern (P', Σ') ("using up" some creases, i.e., $|\Sigma'| < |\Sigma|$) or reports "illegal", as defined below. Let P_l (resp., P_r) be the subset of P in the left (resp., right) open half-plane defined by ℓ. Let Σ_l (resp., Σ_r) be the subset of Σ contained in P_l (resp., P_r). Reflect P_r and Σ_r about ℓ, obtaining P'_r and Σ'_r. If a crease $c \in \Sigma_r$ had an assignment, define $\alpha(c') = -\alpha(c)$ where c' is the reflection of c. Two creases *overlap* if their intersection is a non-degenerate line segment.

Definition 2.1 (Legal folds). *An all-layers simple fold through ℓ is* legal *if it satisfies the following conditions:*

(1) *Line ℓ does not contain any point of a facet of P, i.e., $\ell \cap P \subseteq \Sigma \cup \partial P$, where ∂P represents the boundary of P;*

(2) *No crease in Σ'_r (resp., Σ_l) contains any point of a facet in P_l (resp., P'_r).*

If a (partial) assignment is given, we additionally require the following conditions:

(3) *For every pair of assigned creases c_1 and c_2 in ℓ, $\alpha(c_1) = \alpha(c_2)$.*

(4) *For every pair of assigned creases c_1 and c_2 in $\Sigma_l \cup \Sigma'_r$ that are contained in the transitive closure of the overlap relation (over all creases, both assigned and unassigned), $\alpha(c_1) = \alpha(c_2)$.*

Otherwise we call the fold illegal.

We define the crease pattern (P', Σ') obtained from a legal fold as follows. Let P' be the closure of $P_l \cup P'_r$. Set Σ' initially as $\Sigma_l \cup \Sigma'_r$ and successively merge all pairs of overlapping creases c_1 and c_2. By (2), every interior-intersecting pair of creases overlap. If either c_1 or c_2 is assigned, define the assignment of the new crease as $\alpha(c_1)$ or $\alpha(c_2)$. By (4), all creases can be consistently assigned as above regardless of the order of the merges.

In the one-dimensional case, we define $\mathbb{R}^1$ as the folding plane, P as a line segment and Σ as a set of points in P. A fold is defined by a point ℓ in P, which, by (1), must be also a crease. (3) is vacuously true in the 1D setting. All the other definitions follow by the definitions of the two-dimensional problem.

Finally we can define the problem at hand. The *infinite all-layers simple foldability problem* asks whether, given a crease pattern (P, Σ), there is a sequence of crease patterns $S = ((P_1, \Sigma_1), \dots, (P_m, \Sigma_m))$ such that $(P, \Sigma) = (P_1, \Sigma_1)$, $\Sigma_m = \emptyset$, and (P_i, Σ_i) is the result of a legal infinite all-layers simple fold of (P_{i-1}, Σ_{i-1}) for $i \in \{2, \dots, m\}$.

Difference between definitions. Akitaya et al. [ADK17] defines a flat folded state as an *isometry* of P — an isometric embedding of P into $\mathbb{P}$ that preserves connectivity between facets and creases which imposes the non-stretching restrictions on the paper — together with a non-crossing *layer ordering* — a binary relation between overlapping facets in the isometry describing the above/below relationship between them. Their "non-crossing" definition captures the fact that the paper cannot penetrate itself. A simple fold is an operation that takes a flat folded state and

returns another, modeling the rotation of a portion of the paper through an axis while preserving the non-stretching and non-self-penetrating properties. P can also be defined as an orientable surface with top and bottom sides for which a crease (which can be assigned *mountain* or, resp., *valley*) is the fold through it which brings together the bottom (resp., top) sides of adjacent facets. However, because this paper focuses on only the infinite all-layers model, we propose a simplified definition of simple folds, without defining a folded state. Our definition is equivalent in the sense that it preserves simple foldability of input crease patterns and the sequence S of crease patterns described above can be easily converted into a sequence of flat folded states by defining the isometry based on the sequence of reflections. The layer ordering can be recovered from the assignment of the creases on ℓ (choosing an arbitrary assignment if all creases are unassigned). This conversion is possible because, in the infinite all-layers model, if two facets overlap in the isometry, they can be considered as "glued together" because no simple folds in this model can separate them (in particular, unfolding is not allowed).

3   1D Crease Patterns

In this section, we consider only assigned and unassigned crease patterns. We build on ideas from [ABD$^+$04], representing an instance with n creases as a string S of length $2n+1$, denoting by $S[i], i \in \{1, \ldots, 2n+1\}$ the ith symbol in S. For even i, $S[i]$ represents the assignment of the $(i/2)$nd crease or 0 if unassigned. For odd i, $S[i]$ represents the distance between the $\lfloor i/2 \rfloor$nd and the $(\lfloor i/2 \rfloor + 1)$st crease for odd i, considering the edges of the paper the 0th and $(n+1)$st crease. Let the *complement* of the symbols in the string S be defined as $\mathrm{comp}(S[i]) = S[i]$ (resp., $\mathrm{comp}(S[i]) = -S[i]$) if i is odd (resp., even). This definition is motivated by the observation that, if we consider what happens to the section of paper to the right of crease i when i is folded, we see that the string $S[i+1 \ldots 2n+1]$ is converted to $\mathrm{comp}(S[2n+1 \ldots i+1])$. We show an algorithm that finds the leftmost legal fold defined by the kth crease, if one exists, in time $O(k)$. We then show that successive applications of this algorithm will find a solution for the problem in $O(n)$ time.

The following algorithm finds the smallest prefix of length $2k+1 \leq 2n+1$ of the input S such that: (i) $S[1] \leq S[2k-1]$; (ii) for $i \in [1, k-2]$, then $S[k-i] = \mathrm{comp}(S[k+i])$. Importantly, if conditions (i) and (ii) are satisfied, then a fold through crease k is legal. Note, however, that not every legal fold is necessarily such a satisfying prefix; some may be a satisfying prefix of the *reversed* string S^R (or equivalently, a *suffix* of S). Fortunately, it is easy to handle both cases simultaneously as we will demonstrate in Theorem 1 below.

This algorithm is a minor variation of the algorithm in [Man75] that we show here for completeness. We say that a position $f > k$ fails a crease $S[k]$ if $S[k-(f-k)] \neq \mathrm{comp}(S[f])$, i.e., $(f-k)$ violates (ii) for crease $S[k]$. The array F stores the *failure number* of creases. The failure number $F[k]$ of a crease $S[k]$ is defined as the minimum $f > k$ that fails $S[k]$.
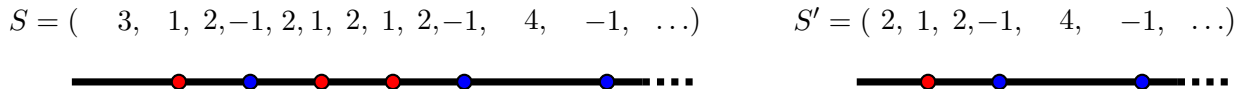
$S = ($ $3,$ $1, 2, -1, 2, 1, 2, 1, 2, -1,$ $4,$ $-1,$ $\ldots)$ $S' = (2, 1, 2, -1,$ $4,$ $-1,$ $\ldots)$

Figure 2: An instance S and the result S' obtained by folding through $S[6]$. A fold though $S[2]$ violates (i), and attempting to fold through $S[4]$ is failed by crease $S[6]$ and, hence, violates (ii).

4

Algorithm A

Data: string S of length $2n + 1$.

1 $k \leftarrow 2$

2 **for** $it \leftarrow 3$ **to** $2n + 1$ **do**

3 $i \leftarrow it - k$

4   **if** $i < 1$ **then**

5   **continue**

    /* Check condition (i)   */

6   **if** $k - i = 1$ **and** $S[1] \leq (S[2k + 1])$ **then**

7   **return** k

    /* Check condition (ii)   */

8   **if** $S[k - i] \neq comp(S[k + i])$ **then**

9 $F[k] \leftarrow i$

10 $k' \leftarrow k + 2$

11   **while** $k' < it$ **do**

12   **if** $it - k' \neq F[k - (k' - k)]$ **then**

13 $F[k'] \leftarrow \min\{it - k', F[k - (k' - k)]\}$

14   **else**

15 $k \leftarrow k'$

16   **go to** 3

17 $k' \leftarrow k' + 2$

18 $k \leftarrow k'$

19 **return** 0.

Lemma 1. *Algorithm A either finds the leftmost legal all-layers simple fold at crease $S[k], 1 < k \leq n + 1$ in $O(k)$ time or returns 0 if no such fold exists in $O(n)$ time.*

Proof. The Lemma follows from [Man75] and [ABD⁺04]. For completeness, we give here a summary of the arguments. Conditions (i) and (ii) for odd (resp., even) indices i imply that a fold at crease $S[k]$ satisfies condition (2) (resp., (3)) from Definition 2.1. Hence, if a value $k \neq 0$ is returned, it defines a legal fold. Similarly, every legal fold at $S[k]$ satisfies conditions (i) and (ii). The integer variables k and it only increase and are upper bounded by $2n + 1$. When a value of k is returned, $it = O(k)$ and, thus, the algorithm runs in $O(k)$ (resp., $O(n)$) time for positive (resp., negative) instances. The algorithm maintains the invariant that every crease to the left of k have been assigned a failure number, i.e., there exists no legal fold to the left of k.

The algorithm reaches line 9 if position it fails $S[k]$. For all creases $S[k']$ where $k < k' < it$, we can determine in constant time if a position to the left of it fails $S[k']$ as follows. Let $S[k'']$ be the crease symmetric to $S[k']$ about $S[k]$, i.e., $k'' = k - (k' - k)$. If $F[k''] < it - k'$, then by (ii) with respect to $S[k]$ we have $S[k'' + F[k'']] = comp(S[k' - F[k'']])$ and $S[k'' - F[k'']] = comp(S[k' + F[k'']])$. By the definition of failure number, $F[k'']$ is the smallest integer such that

$$comp(S[k' - F[k'']]) \neq comp(comp(S[k' + F[k'']])) = S[k' + F[k'']].$$

Notice that $a = comp(comp(a))$ and $a \neq comp(b) \leftrightarrow comp(a) \neq b$ for all a and b. Hence,

$F[k'] = F[k'']$ and a fold through $S[k']$ is illegal. If $F[k''] > it - k'$, then $S[k'' + (it - k')] =$ comp$(S[k'' - (it - k')]) = S[k' - (it - k')]$. Because the position it fails $S[k]$,

$$S[k' + (it - k')] \neq \text{comp}(S[k'' - (it - k')]) = \text{comp}(S[k' - (it - k')]).$$

Therefore, $F[k'] = it - k'$ and a fold through $S[k']$ is illegal. Then, the algorithm finds the leftmost crease k for which no position between k and it fails $S[k]$. Therefore, the algorithm returns the leftmost legal crease on the left half of S. □

Theorem 1. *All-layers simple foldability of a 1D assigned or unassigned crease patterns can be decided in deterministic linear time.*

Proof. We provide a constructive proof. Run Algorithm A on inputs S and S^R, a reversed copy of S, in parallel. If both return 0, then by Lemma 1, S is not all-layers simple foldable. Else, let k be the first value returned. Without loss of generality, k was returned by Algorithm A on S. Using our definitions for folding at $S[k]$, the resulting crease pattern S' is represented by the substring of S from $k + 1$ to $2n + 1$. Hence we generated a smaller subproblem of size $2n - k$ in $O(k)$ time. By Lemma 4.1 of [ABD$^+$04], if any legal fold can be done in S, the resulting crease pattern S' is all-layers simple foldable if and only if S also is. Then, by successively applying the above algorithm in the resulting smaller subproblems we arrive at a crease pattern $(P_m, \emptyset)$ in $O(n)$ time if and only if S is all-layers simple foldable. □

Combining the theorem above with Theorem 5.2 of [ABD$^+$04], we can solve axis-aligned orthogonal crease patterns on rectangular paper in the same running time:

Corollary 1. *All-layers simple foldability of a 2D assigned or unassigned crease pattern with axis-aligned creases on axis-aligned rectangular paper can be decided in deterministic linear time.*

4   Unassigned Orthogonal 2D Crease Patterns

Let (P, Σ) be the input for our problem, where P is an orthogonal polygon and Σ contains axis-aligned creases. We first show a necessary condition that any crease pattern of a solution $((P_1, \Sigma_1), \ldots, (P_k, \emptyset))$ must satisfy.

Lemma 2. *Given a crease pattern (P, Σ) with orthogonal paper P and orthogonal creases Σ, if a line ℓ^* in $\mathbb{P}$ contains a crease in Σ, then every point in $P \cap \ell^*$ must be on a crease in Σ or else (P, Σ) is not infinite all-layers simple foldable.*

Proof. Recall that, by condition (1), a simple fold through a line ℓ does not contain any point of a facet. By contradiction, let ℓ^* be a line containing a crease such that $P \cap \ell^* \setminus \partial P$ is not covered by creases in Σ. We prove by induction that (P, Σ) is not simple foldable. Trivially, ℓ^* cannot be an axis of a simple fold by (1). The base case is when all creases in $\Sigma \neq \emptyset$ are on an axis that define illegal simple folds, hence, no legal fold is possible. Else, a legal fold through some axis ℓ is possible. By (2), a simple fold maps points on creases to creases and facet points to facet points. If ℓ^* is parallel to ℓ, the creases and facet points on ℓ^* will be mapped to a creases and interior points on a line ℓ' ($\ell' = \ell^*$ if ℓ is to the left of ℓ or ℓ' is the reflection of ℓ^* about ℓ). Else, ℓ^* is perpendicular to ℓ and ℓ^* will also contain a crease and a facet point of (P', Σ') after the fold. Because $|\Sigma'| < |\Sigma|$, we are done. □

6

We now describe the encoding of the input (P, Σ). Let n_P be the number of vertices in P. Let $L = \{\ell_1, \ldots, \ell_{n_L}\}$ be the set of lines in $\mathbb{P}$ that contain creases in Σ. By Lemma 2, it is enough to store L because Σ can be inferred by L and P or else (P, Σ) is a negative instance. Our algorithm uses (P, L) as the input and define its size as $n_P + n_L$. First, we provide an algorithm for when P is a rectangle.

Lemma 3. *Let (P, L) be an unassigned crease pattern where P is a rectangle and lines in L are axis-aligned. Infinite all-layers simple foldability of (P, L) can be determined in linear time.*

Proof. We reduce the problem to two 1D instances, each solvable in $O(n')$ time where n' is the number of creases in the 1D instance by Lemma 1. Let L_h (resp., L_v) be the subset of L containing horizontal (resp., vertical) lines. Build a 1D instance with paper P_v (resp., P_h) being a line segment congruent to the left (resp., bottom) edge of P. Add a crease in P_v (resp., P_h) at the intersections between the lines in L_h (resp., L_v) and its correspondent edge. By definition, any legal fold though a line in L_h (resp., L_v) is also a legal fold in its corresponding crease in P_v (resp., P_h), and its resulting crease pattern (P', L') can be converted into 1D problems P'_v and P'_h such that P'_v (resp., P'_h) is equal to the resulting crease patterns of the fold in P_v (resp., P_h), and $P_h = P'_h$ (resp., $P_v = P'_v$). Therefore, any folding sequence of the instance (P, L) satisfying the problem can be converted to two folding sequences of the 1D problems and vice versa. $\square$

Theorem 8 of [ADK17], which says that instances of simple foldability with rectangular paper and orthogonal creases are equivalent in the all-layers model and infinite all-layers model, implies that the reduction in Lemma 3 also applies to the all-layers model. Now, we prove our algorithmic result.
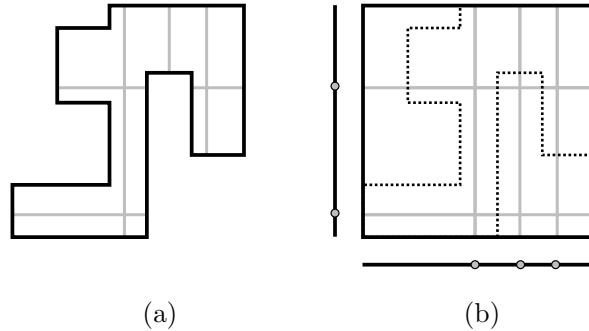

(a)   (b)

Figure 3: (a) An instance (P, L) where P is an orthogonal polygon and (b) its corresponding instance (R, L) where R is the bounding box of P and the boundary of P is shown with dotted line segments. The 1D instances corresponding to (R, L) are also shown in (b).

Theorem 2. *Determining whether an unassigned crease pattern on orthogonal paper with orthogonal creases has an infinite all-layers simple folding can be solved in linear time.*

Proof. We reduce to the rectangular paper/orthogonal creases problem solved in Lemma 3. Let the instance with orthogonal paper and creases be represented by (P, L) of size $n = n_1 + n_2$ where n_1 and n_2 are the sizes of P and L, respectively. We can obtain the bounding box R of P in $O(n_1)$ time. Output the instance (R, L) that can be solved in $O(n_2)$ time by Lemma 3. We now show that (R, L) admits an infinite all-layers simple folds sequence if and only if (P, L) also does.

The backward implication is straightforward. If (R, L) has a simple folding, then so must (P, L). Notice that creases induced by L in P are a subset of the creases induced by L in R. Because P is a subset of R, by definition, any fold made on R will also be possible on P, so P is infinite all-layers simple foldable.

Now we show that, if $((P_1, L_1), \ldots, (P_k, L_k))$ is an infinite all-layers simple folding for (P, L), then $((R_1, L_1), \ldots, (R_k, L_k))$ is an infinite all-layers simple folding for (R, L), where R_i is the bounding box of P_i for $i \in \{1, 2, \ldots, k\}$. We show that the axis of the simple fold from (P_i, L_i) to (P_{i+1}, L_{i+1}) is also the axis of a simple fold that transforms (R_i, L_i) into (R_{i+1}, L_{i+1}). By definition, P_i and R_i share at least one each of bottom, left, top, and rightmost points, which we denote by x_b, x_l, x_t and x_r respectively. Without loss of generality, take a vertical axis ℓ pointing up whose supporting line is in L_i that defines a legal fold. By definition, the simple fold through ℓ reflects the right portion of P_i and R_i to the left of ℓ producing P_{i+1} and R'_i. Lemma 2 guarantees that both originate the same set of new supporting lines L_{i+1}. Additionally, in both cases, the rightmost point of the paper after the fold will be on ℓ. If the reflection of x_r becomes the leftmost point in P_{i+1} it will also become the leftmost point of R'_i. Because x_t and x_b don't change y-coordinates, $R'_i = R_{i+1}$. $\qquad \square$

5   Partially Assigned Orthogonal Crease Patterns

We prove that, given a crease pattern with a subset of creases assigned mountain/valley, it is NP-complete to decide infinite all-layers simple foldability, even if the crease pattern is a square grid and the paper is rectangular.

Lemma 4. *It is strongly NP-hard to decide infinite all-layers simple foldability of partially assigned crease patterns, even if the paper is rectangular and the creases form a square grid.*

Proof. We provide a reduction from 3SAT. An instance of 3SAT is given by a set of n variables $\{x_1, \ldots, x_n\}$ and a boolean formula in conjunctive normal form with m clauses of the form $(l_1 \vee l_2 \vee l_3)$ where l_k, $k \in \{1, 2, 3\}$, is the boolean value of a variable x_i, $i \in \{1, \ldots, n\}$, or its negated value denoted by $\overline{x_i}$. The 3SAT problem asks whether there exists an assignment from the variables to $\{\text{TRUE}, \text{FALSE}\}$ such that the boolean formula evaluates to TRUE. Given the 3SAT instance described above, we construct a crease pattern on a $2n(5m + 1) \times (2n + m + 3)$ rectangular paper as shown in Figure 4 (a).

Overview. Refer to Figure 4 (a). Informally, a variable x_i is represented by the section of paper between rows f_i and t'_i inclusive. The yellow dots represent literals in a clause and folding a yellow dot onto a green or red dot represents assignment as positive or negative respectively. Yellow dots (literals) sharing a vertical line are in the same clause. Note that at first the only line through the paper that has "consistent" assignment are t_1 or f_1. Folding through any other line would violate (4). The assigned creases on vertical lines v_i and v'_i form a staircase on the upper right corner, and the staircase's complement alternates streaks of mountain and valley. This staircase forces any sequence of legal simple folds to choose between folding through t_1 or f_1, bringing them on top of a green or red dot, respectively, which encodes the boolean assignment of the variable x_1. After a vertical fold on the right edge of the paper, a similar argument forces these yellow dots to coincide with the yellow dot right above it by folding through t'_1 or f'_1, which adds a valley assignment to a crease next to a yellow dot of a corresponding clause if and only if its literal of x_1 evaluates to

FALSE; see Figure 4 (b–d). We apply induction on the resulting crease pattern and, after choosing the assignment of all variables, the topmost set of yellow points will have at least one crease that is not assigned valley if and only if the SAT instance has a positive solution. If a vertex has all four incident valley-assigned creases, there is no legal simple fold that folds through any of its creases.

Construction. Consider the bottom left corner as the origin. In the following, we use the indices $i \in \{1, \ldots, n\}$ to refer to variables and $j \in \{1, \ldots, m\}$ to refer to clauses. We first define the position of some points that will serve as reference to the construction. For each clause c_j, define the points $p_{j,i} = (j + 1, 10mi - 5j - 5m + 2i)$ and a point $p_{j,(n+1)}$ using the same formula, that represent $n + 1$ copies of the clause. In Figure 4 (a), such points are represented as yellow dots. Assign the crease right below each $p_{j,i}$ as valley. Also define $r_{j,i} = (j + 1, 5j - 10m + i(10m + 2) + 6)$ and $g_{j,i} = r_{j,i} + (0, 2)$, represented in Figure 4 (a) as red and green dots respectively. If the first (resp., second, resp., third) literal that c_j contain is x_i, assign the crease to the right (resp., bottom, resp., left) of $r_{j,i}$ as mountain. If the literal is negated, i.e., $\overline{x_i}$, assign the crease to the right (resp., bottom, resp., left) of $g_{j,k}$ as mountain.

We now define some reference horizontal lines. Let f_i be the horizontal line appearing at height $5m + 2(i - 1)(5m + 1)$, and t_i, f_i' and t_i' the horizontal lines 1, $5m + 1$ and $5m + 2$ units above f_i respectively. Define f_{n+1} and t_{n+1} analogously. For each variable x_i, we define the crease assignment of two vertical lines v_i and v_i', which are $m + 2n - 2i + 2$ and $m + 2n - 2i + 3$ units to the right of the origin respectively. Also define v_{n+1}', which is $m + 1$ units to the right of the origin. Assign $5m$ creases bellow f_k as valley for $k \leq i$. Assign $5m - 2$ creases bellow t_k' as mountain for $k < i$, skipping the first one. Assign mountain to $5m - 1$ creases below f_i' in v_i and to all the creases above t_i in v_i'. Assign valley to all the creases above t_i' in v_i. Finally, we describe the assignment of creases in f_{n+1}. Assign the $m + 2$ leftmost creases as valley and alternate between mountain and valley until the end of the line. All other creases in the construction that do not have any assignment remain unassigned.

Correctness. We first show that, if the 3SAT instance has a positive solution, so does the constructed instance of simple foldability. In order from 1 to n, if x_i is assigned TRUE (resp., FALSE), valley fold through t_i (resp., f_i), bringing the bottom part up and aligning all $p_{j,i}$ with $g_{j,i}$ (resp., $r_{j,i}$); mountain fold though v_i', bringing the right strip of paper to the left; valley fold through t_i' (resp., f_i'), aligning all $p_{j,i}$ with $p_{j,i+1}$; valley fold through v_i. If x_i appears as the first (resp., second, resp., third) literal of c_j which evaluates to FALSE, then these folding sequence brings a valley-assigned crease to the right (resp., top, resp., left) of $p_{j,i+1}$; see Figures 4 (b) and (d). After all n steps, valley fold through f_{n+1}, bringing the top strip of paper down, and then through v_{n+1}'. This eliminates all assigned creases apart from the ones adjacent to reference points. Notice that, for a clause c_j, the corresponding assigned creases (which are adjacent to a reference point with index j) are contained in a 2×5 bounding box containing $p_{j,n+1}$ in its $(1, 2)$ relative coordinate (shown as a dotted orange box in Figure 4 (a) and Figure 5). All boxes containing $p_{k,n+1}, k < j$ are below and to the right of the point $(0, -2)$ relative to $p_{j,n+1}$ (darkened region shown in Figure 5). We start by assuming that all folds relative to $c_k, k < j$ are already folded and the top left corner of the paper coincides with a corner of the box containing $p_{j,n+1}$. Recall that if the first (resp., second, third) literal of c_j evaluates to FALSE, the left (resp., top, right) crease incident to $p_{j,n+1}$ will be assigned valley. Else, a crease two units above or below will be assigned valley, depending on whether the literal is negated or positive. If the second literal of c_j evaluates to TRUE, the assigned creases in the bounding box of $p_{j,n+1}$ are a subset of the leftmost crease pattern in Figure 5 (a). If both the first and third literal evaluate to TRUE, then the assigned creases in the bounding box of

$p_{j,n+1}$ are a subset of the pattern in Figure 5 (b). We can fold through all creases in such box using the folding sequence in Figure 5 (a) (resp., (b)), which does not introduce any assignment in boxes of $p_{l,n+1}$, $l > j$. Else, the assigned creases are a subset of the leftmost crease pattern in Figure 5 (c), or a subset of the crease patterns in Figure 5 (d). By following the folding sequence in Figure 5 (c), we eliminate all assigned creases in the box of $p_{j,n+1}$ while not introducing any assignment in boxes of $p_{l,n+1}$, $l > j$. By induction, we can continue folding until there are no assigned creases and the resulting crease pattern has a positive solution using the algorithm for unassigned crease patterns.

Now, we show that if the produced crease pattern is foldable, the 3SAT instance has a positive solution. By (4), no vertex can have 4 adjacent creases with the same assignment or else there is no legal fold through such vertex. Initially, the only possible legal folds are through f_1 or t_1 due to (4). First, assume that the solution folds through f_1, bringing $p_{j,1}$ onto $r_{j,1}$; see Figure 5 (b). That assigns some creases between f_1 and t_1 making it impossible to fold through t_1. The only possible simple fold now is through v'_1. Then, folding through t'_1 is illegal since it would bring a mountain-assigned crease in v'_2 right above f_1 onto a mountain-assigned crease right above t_2; see Figure 5 (c). Hence, the only possible fold is through f'_1, followed by a valley fold through v_1. The resulting folded state has a height-1 strip of paper in the bottom that does not contain any assigned crease. If this is folded on top of any other part of the construction, it will not create any other assignment. At this state, the solution can either fold through the horizontal line right above t'_1 and then fold through f_2 or t_2, or fold directly through f_2 or t_2, which are the only available legal folds. In both scenarios, the resulting assigned pattern and the boundary of the paper are the same. Therefore, after folding through v_1, we obtain a smaller version of the reduction with variables $\{x_2, \ldots, x_n\}$ and some extra assigned creases in the vicinity of $p_{j,2}$ if c_j contained a literal of x_1. In particular, if said literal is positive, there exists one extra crease adjacent to $p_{j,2}$ that is assigned valley. Now, assume that the solution folds through t_1. The next folds must be through v'_1 followed by t'_1, and v_1 or else two valley crease will be mapped onto each other contradicting (4); see Figure 5 (d). We again obtain a smaller version of the reduction and if c_j contained the literal $\overline{x_1}$, there is one extra crease adjacent to $p_{j,2}$ that is assigned valley. By induction, the solution must fold in this manner (bringing $p_{j,i}$ onto $p_{j,i+1}$) until all of the yellow marked points are on top of a $p_{j,n+1}$. Because the crease pattern in foldable, there must exist at least one crease adjacent to $p_{j,n+1}$ that is not assigned valley. That corresponds to an assignment in which at least one literal in each clause evaluates to TRUE. □

Theorem 3. *It is strongly NP-complete to decide simple foldability under the infinite all-layers model of a partially assigned crease pattern, even if the paper is rectangular and the creases form a square grid.*

Proof. It remains to show that the problem is in NP. Because unfolding is forbidden, a simple fold reduces the number of creases by at least one. Therefore, a sequence of simple folds that folds through all creases of a crease pattern with n creases can have $O(n)$ folds. We can check in $O(n)$ time whether a given axis defines a legal simple fold. The rest of the proof follows from Lemma 4. □

6 Open Problems

Lemma 4.1 of [ABD$^+$04], used in Theorem 1, only applies to assigned or unassigned 1D crease patterns. Hence, we leave open the algorithmic complexity of deciding all-layers simple foldability

of partially assigned 1D crease patterns. Indeed, for this class of input, the problem has not been studied in the other simple foldability models introduced in [ABD+04]. We conjecture that the algorithm presented in Section 3 also finds a solution if one exists. The main remaining open problem is whether infinite all-layers simple foldability can be solved in polynomial time, or is NP-hard, in fully assigned crease patterns [ADK17]. This problem remains open in particular when restricted to orthogonal crease patterns. We also leave open the complexity of infinite all-layers simple foldability in unassigned nonorthogonal crease patterns, for example, axis-aligned rectangular pieces of paper with unassigned creases at multiples of $45°$, or even general unassigned crease patterns on general polygonal pieces of paper.

Acknowledgments

References

[ABD+04] Esther M. Arkin, Michael A. Bender, Erik D. Demaine, Martin L. Demaine, Joseph S. B. Mitchell, Saurabh Sethia, and Steven S. Skiena. When can you fold a map? *Computational Geometry: Theory and Applications*, 29(1):23–46, 2004.

[ACD+15] Hugo A. Akitaya, Kenneth C. Cheung, Erik D. Demaine, Takashi Horiyama, Thomas C. Hull, Jason S. Ku, Tomohiro Tachi, and Ryuhei Uehara. Box pleating is hard. In *Revised Papers from the 18th Japan Conference on Discrete and Computational Geometry and Graphs*, pages 167–179, 2015.

[ADK17] Hugo A. Akitaya, Erik D. Demaine, and Jason S. Ku. Simple folding is really hard. *Journal of Information Processing*, 25:580–589, 2017. Special issue of papers from the 19th Japan Conference on Discrete and Computational Geometry, Graphs, and Games, September 2016.

[BH96] Marshall Bern and Barry Hayes. The complexity of flat origami. In *Proceedings of the 7th Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 175–183, 1996.

[BM08] Devin J Balkcom and Matthew T Mason. Robotic origami folding. *The International Journal of Robotics Research*, 27(5):613–627, 2008.

[DM01] Erik D. Demaine and Joseph S. B. Mitchell. Reaching folded states of a rectangular piece of paper. In *Proceedings of the 13th Canadian Conference on Computational Geometry*, pages 73–75, Waterloo, August 2001.

[Man75] Glenn Manacher. A new linear-time "on-line" algorithm for finding the smallest initial palindrome of a string. *J. ACM*, 22(3):346–351, July 1975.
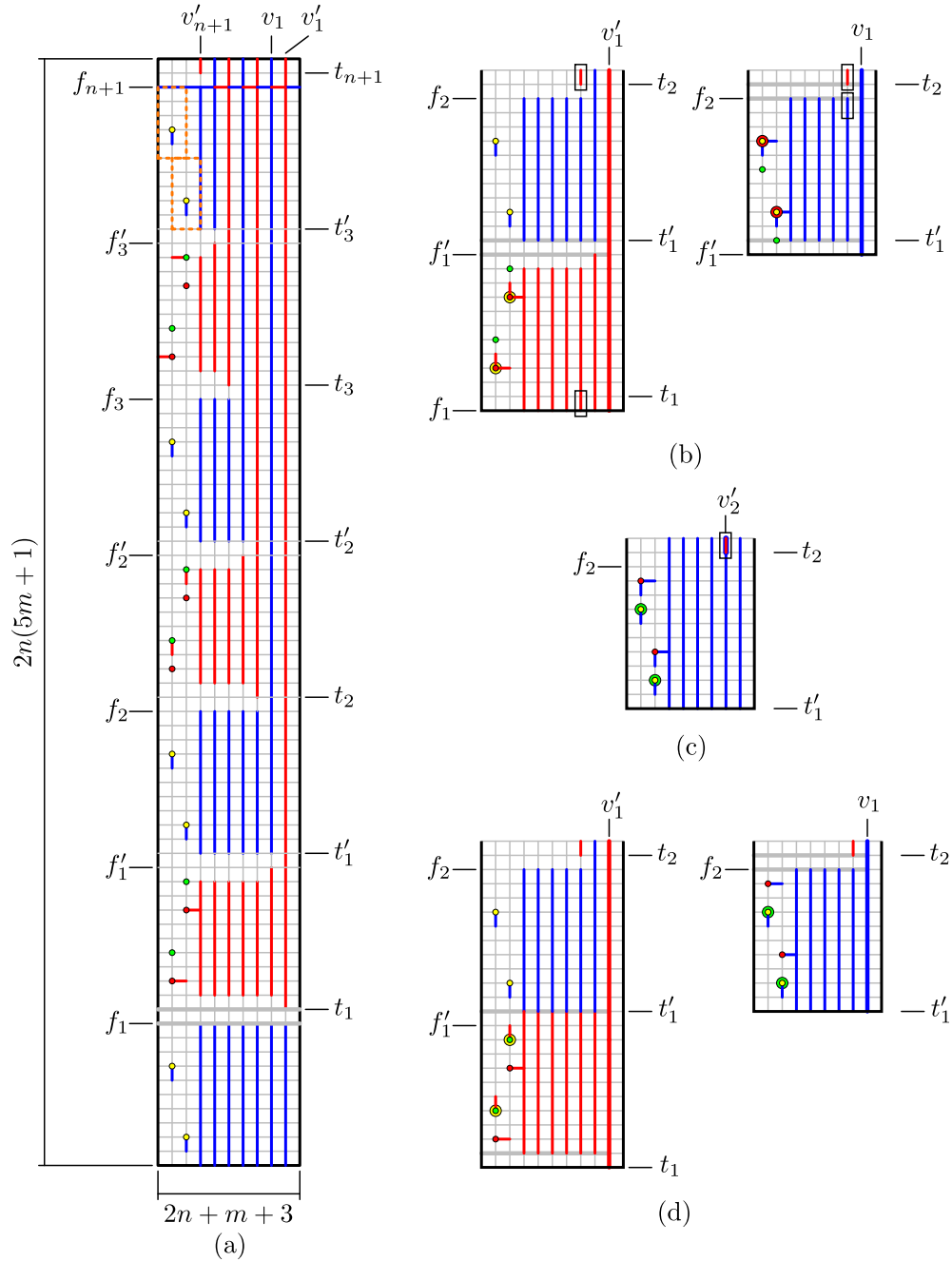
Figure 4: (a) Reduction from 3SAT to partially assigned simple foldability under the infinite all-layers model. The instance shown corresponds to the boolean formula $(x_1 \vee \overline{x_2} \vee x_3) \wedge (x_1 \vee \overline{x_2} \vee \overline{x_3})$. The only possible initial folds are through f_1 or t_1 (shown in bold). (b) The resulting crease pattern obtained by folding (a) through f_1 (left) and then through v'_1 and f'_1. The yellow dots are brought on top of the red dots, and these are brought on top of the next set of yellow dots. (c) Attempting to fold the pattern in the left part of (b) through v'_1 and then through t'_1 violates (4). (d) The resulting crease pattern obtained by folding (a) through t_1 (left) and then through v'_1 and t'_1 (right). The yellow dots are brought on top of the green dots, and these are brought on top of the next set of yellow dots.
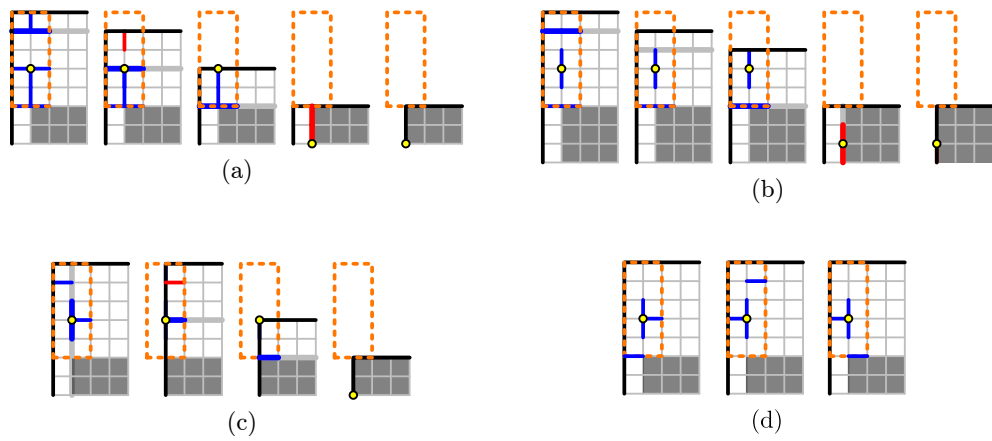
12

Figure 5: Folding sequences that eliminate all assigned creases in the vicinity of $p_{j,n+1}$ for a positive instance after folding through f_{n+1}. A subset of the blue line segments shown in the leftmost crease pattern of (a–b) will be assigned valley if (a) the second literal evaluates to TRUE, or (b) the first and third literals evaluate to TRUE. (c) If neither (a) or (b) occur and the first literal is nonnegated and evaluates to TRUE. (d) Remaining cases for a satisfied clause that can be folded using the same folding sequence shown in (c).