Unfolding Rectangle-Faced Orthostacks

Erin W. Chambers* Kyle A. Sykes† Cynthia M. Traub‡

Abstract

We prove that rectangle-faced orthostacks, a restricted class of orthostacks, can be grid-edge unfolded without additional refinement. We prove several lemmas applicable to larger classes of orthostacks, and construct an example to illustrate that our algorithm does not directly extend to more general classes of orthostacks.

1 Introduction

An *unfolding* of a polyhedron is a cutting of the surface of the polyhedron such that the surface may be unfolded into the plane as a simple polygon where the interior of any two faces does not overlap. An *edge unfolding* considers only cuts made along edges, while *general unfoldings* allow cuts anywhere on the surface.

There are many open questions relating to polyhedral unfoldings. For example, while it is known that not every nonconvex polyhedron has an edge-unfolding, it is still open whether every polyhedron has a general unfolding. In general, progress has been made on this problem by considering restricted classes of polyhedra [2, 3, 10, 5] or by varying the type of cuts that are allowed, such as vertex unfoldings [7, 6, 4] or star unfoldings [1, 9]. See [8, 11] for surveys of this area.

We will consider an unrefined grid-edge unfolding of a class of axis-orthogonal polyhedra known as orthostacks (formally defined in Section 2). An unrefined *grid-edge* unfolding creates new edges on the surface of a polyhedron by intersecting the surface with planes parallel to the $x, y, z-$axes through every vertex of the polyhedron. Any of the edges from the original polyhedral surface as well as these new edges are now available for cutting. The technique of grid-edge refinement can be generalized by further dividing every rectangle of the surface into $k \times l$ rectangles. An unrefined grid-edge unfolding is thus a 1×1 refinement. It is known that every orthostack can be grid-edge unfolded with a 1×2 refinement [2]. In this paper we prove that a certain class of orthostacks (which we call "rectangle-faced orthostacks") can be grid-edge unfolded without further refinement of the surface.

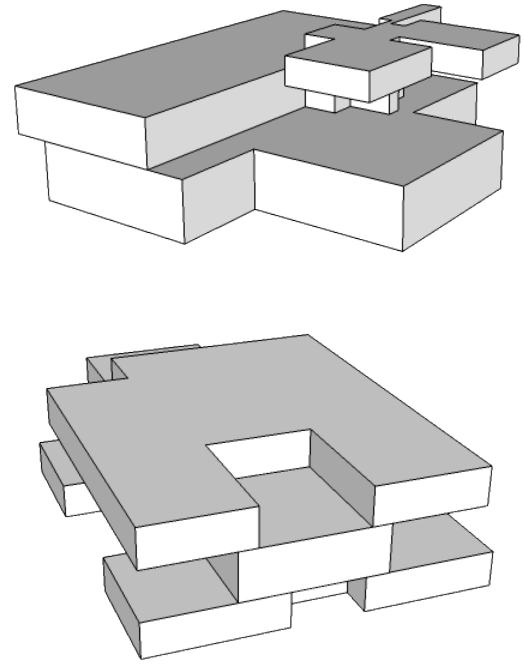

Figure 1: (a) An example of an orthostack. (b) A rectangle-faced orthostack.

Our algorithm is a natural one given the structure of orthostacks, where we unfold each layer of the orthostack and connect them via "bridges" between the layers. It has a similar setup to the one for the 1×2 refinement [2], although theirs cannot choose bridges in the same fashion; they instead cut each band vertically in half to "shift" the bridge-rectangle to the top position. Unfortunately, our algorithm will not extend to general orthostacks; in section 5 we present a (non-rectangular-faced) orthostack which our algorithm fails to unfold. We conclude with a discussion of how our structural results may be useful for computing 1×1 unfoldings of more general classes of orthostacks.

2 Definitions

An orthostack P is a genus-zero axis-orthogonal polyhedron with the property that that in at least one dimension, each distinct cross section of P is an orthog-

*Department of Mathematics and Computer Science, Saint Louis University, echambe5@slu.edu. Research supported in part by the NSF under Grant No. CCF 1054779.

†Department of Mathematics and Computer Science, Saint Louis University, ksykes2@slu.edu

‡Department of Mathematics and Statistics, Southern Illinois University Edwardsville, cytraub@siue.edu

onal polygon that is both connected and simply connected (containing no holes). Without loss of generality, assume this dimension is the $z-$dimension. Following the terminology used in [2], we name the faces of the orthostack according to the axis to which they are orthogonal, thus giving rise to $x-$, $y-$ and $z-$faces. Changes in the $z-$cross sections of an orthostack occur at finitely many $z-$coordinates $z_0, z_1, \ldots, z_k$. We call the *i-th band* B_i the collection of $x-$ and $y-$faces that form the boundary of the orthostack for $z_{i-1} \leq z \leq z_i$, where i ranges from 1 to k. So the faces of the orthostack P are therefore partitioned into bands B_i $(1 \leq i \leq k)$ and the $z-$faces that occur in the planes $z = z_0, z = z_1, \ldots, z = z_k$. Note that the $z-$faces occur at the "top" and "bottom" of the orthostack as well as in the layers forming transitions between bands. We will also let *layer* L_i be the subset of the orthostack with $z_{i-1} \leq z \leq z_i$, which is the 3-dimensional solid bounded by $B_i, z = z_{i-1}$, and $z = z_i$.

We call an orthostack *rectangle-faced* if every $z-$face is a rectangle, excluding the $z-$faces on the top of bottom of the orthostack, and edges of these z-faces are entirely along one band or another (with no edge belonging to both adjacent bands). Examples of orthostacks with and without the rectangle-faced property are shown in Figure 1.

3 Structural results

We begin with several structural lemmas regarding orthostacks. Note that these results apply to *any* orthostack, not just rectangle-faced ones, and may be of use for more general classes of orthostacks.

Lemma 1 *Any z-face at height* $z = z_i$, $1 \leq i \leq k-1$, *must be incident to both* B_i *and* B_{i+1}.

Proof. Suppose to the contrary that some z-face, R, at height $z = z_i$ has edges only incident to one band, which we assume without loss of generality is B_i. The face R does not occur at $z = z_0$ or $z = z_k$ due to our initial assumption on i, so there must exist a subset of $B_i \cap (z = z_i)$ that is not incident to R. (Else, the orthostack will not continue past the face R, a contradiction.) The intersection produced by slicing the orthostack with a plane $z = z_i - \epsilon$ for a sufficiently small value of ϵ will either be disconnected or a degenerate polygon consisting of (at least) two polygons attached at a single vertex. Both situations contradict the definition of an orthostack, since each z-slice must be a simply connected polygon. Therefore, R must have edges incident to both B_i and B_{i+1}. □

Lemma 2 *The perimeter of any z-face at height* $z = z_i$, $1 \leq i \leq k-1$, *is partitioned into two contiguous components, one incident to band* B_i *and the other incident to* B_{i+1}. *Moreover, some pair of opposite edges a and b of the face will have edge* e_1 *containing a segment incident to* B_i *and edge* e_2 *containing a segment incident to* B_{i+1}.

Proof. Assume that the boundary of a z-face R is partitioned into more than two contiguous components from bands B_i and B_{i+1}; see Figure 2. At this z_i layer, the rectangle must be visible; this happens due to a change in the layers of the orthostack. Namely, the cross-sections above and below $z = z_i$ are distinguished by which cross-section includes R. Thinking of R in terms of x and y coordinates, we assume without loss of generality that, for sufficiently small ϵ, $R \times [z_i - \epsilon, z_i]$ is contained in layer L_i and $R \times [z_i, z_i + \epsilon]$ is exterior to layer L_{i+1}.

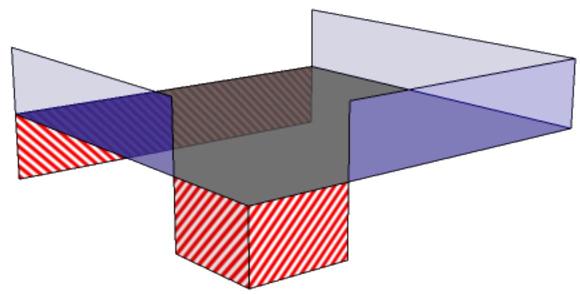

Figure 2: A 3-dimensional view of how rectangle R (shaded darker) appears in the orthostack; the adjacent portions of B_i and B_{i+1} that border R are shown red (striped) and blue (solid).

Since each intersection of the orthostack with a z-plane is one simply connected polygon, the two or more connected components of $R \cap B_{i+1}$ must be pathwise connected to one another via $B_{i+1} \cap (z = z_i)$, a curve we color solid blue in Figure 3. Moreover, since the blue curve is the boundary of a simply connected polygon in the $z = z_i$ plane, it will not self intersect.

Two possible orientations of these paths are given in Figure 3, where the rectangle boundaries shaded in red come from band B_i and those from B_{i+1} are shaded in blue. In case 1 (Figure 3, left), there is a region (illustrated near the upper right corner of the rectangle) exterior to the cross section of the rectangular face but completely surrounded by the union of the rectangle with the area bounded by the solid blue curve. This forms a cross-section at $z = z_i$ in the orthostack which fails to be simply connected, contradicting the definition of orthostack. In case 2 (Figure 3, right), the blue curve represents an "inner" boundary of B_{i+1}, which means that the cross section at $z = z_i$ will also fail to be simply connected (since there is a gap between the rectangle and the blue curve marking the inner boundary

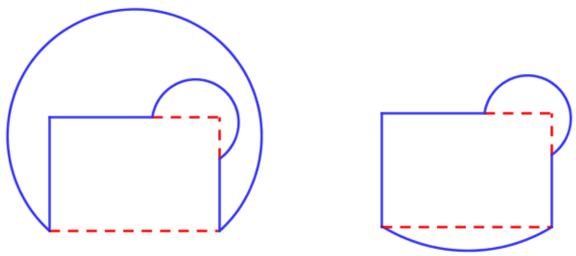

Figure 3: Two possible visualizations for our face R, where the red (dashed) component of the boundary is adjacent to band B_i and the blue (solid) is adjacent to band B_{i+1}.

of B_{i+1}). In either case, we contradict the assumption that the original object was an orthostack and therefore had simply connected orthogonal cross sections.

Note that in either of the two cases, the contradiction appears between two components of the boundary of R induced by B_{i+1} that appear sequentially around the boundary of R. While our image only shows 2 connected components total, the same contradiction is present even if more than two connected components are present along the boundary of R. □

We will use lemma 2 in the next section to set up "bridges" between the unfolded bands in our algorithm, but we also need to characterize the order in which the z-faces are encountered before proceeding with our algorithm.

Lemma 3 *For any orthostack, the cyclic ordering of the rectangles at height $z = z_i$, given by tracing around the band B_i counterclockwise and numbering z-faces by the order in which they are encountered, is the same as the cyclic ordering given by tracing the band B_{i+1} counterclockwise.*

Proof. Begin by fixing $1 \leq i \leq k-1$. First suppose that the bands B_i and B_{i+1} do not share any common point. If there are any z-faces, then using the previous lemma, we know that the z-face must form an annulus bounded by B_i on one side and B_{i+1} on the other, since any other configuration will either result in a common point or non-contiguous components adjacent to B_i or B_{i+1}. Moreover, this must be the only z-face at this level, since otherwise the band must stop tracing the boundary of the z-face and later re-enter it after tracing around another z-face, which results in that layer not being simply connected. Since there is only one z-face, the statement of the lemma holds trivially.

Now if B_i and B_{i+1} have some common point, pick any one of them as a start point. Proceed counterclockwise on a path along the boundary at cross section z_i shared by both bands until the bands diverge, which must happen if there is any z-face at height z_i. When the path diverges, you have met a z-face α at a boundary point where B_i and B_{i+1} meet. One path will trace the boundary of α shared with B_i and the other will trace the boundary of α shared with B_{i+1}. Note that neither band can move away from α and then return, since by Lemma 2 we know that each band stays adjacent to the face in a single connected component along the boundary of the α. Label this z-face as α_1. The paths continue tracing their respective boundaries until they intersect at a point shared by both boundaries, which again follows from Lemma 2. At this point, the paths merge into a single path again and continue tracing counterclockwise along a portion adjacent to both bands (which possibly consists of only a single point, if we meet the next z-face immediate). We can continue following the bands counterclockwise along the shared boundary at height z_i, with the bands diverging only when they meet the same new z-face. We will label these z-faces in the order we meet them. Proceeding in this manner around the perimeter of both bands gives a unique labeling of all the z-faces as $\alpha_1, \alpha_2, \ldots, \alpha_k$ which is common to both bands, so the clockwise ordering is identical. □

4 Algorithm

In this section, we restrict our attention to *rectangle-faced orthostacks*, where the z-faces are rectangles whose edges entirely belong to the boundary of B_i or B_{i+1}, but not both.

Our unfolding algorithm proceeds as follows. We will unfold into an xz-plane, in order of increasing z-coordinate. References to x-coordinates will refer to the unfolded shape in this plane. Start by unfolding band B_1 arbitrarily. Attach the z_0-face arbitrarily below band B_1. Loop through the following steps for $i = 1$ to $k-1$.

1. Consider all the rectangles from layer $z = z_i$. We know these must attach to both B_i and B_{i+1} on some opposite pair of edges by Lemma 2. We consider only these opposite edges as possible attachments for these z-faces, and attach them arbitrarily to the band B_i.

2. Now among the attached rectangles, choose the one which has the highest x-coordinate; this is our "bridge" between B_i and B_{i+1}. We then attach band B_{i+1} to the bridge rectangle and unfold B_{i+1} "upward" into the increasing x direction.

Lastly, we can glue z_k-face to the final band arbitrarily, since there is no "next" band to conflict with any possible attachment point.

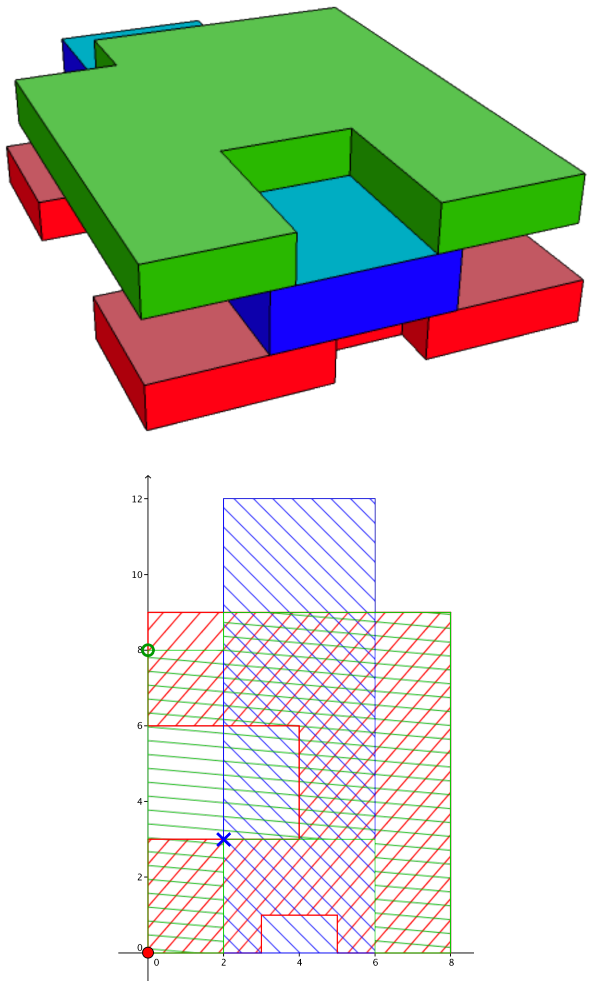

Figure 4: Top: An example of a rectangle-faced orthostack. Bottom: The same orthostack when viewed looking towards the $-z$ direction. A solid red dot indicates where we choose our initial cut for band B_1 (Red, bottom layer in top picture), a blue X indicates where the band B_2 (Blue, middle layer) is cut, and the hollow green point indicates where the band B_3 (Green, top layer) is cut.

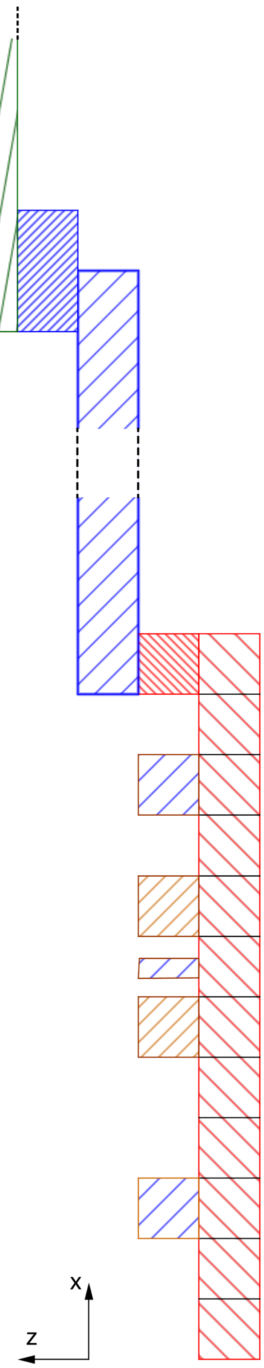

Figure 5: An partial unfolding of the orthostack in Figure 1(b). The bands are not to scale, and only the $z-$faces where $z = z_1$ between B_1 and B_2 are shown attached to B_1.

5 Conclusion & Further Work

It remains to be shown whether every orthostack can be grid-unfolded with a 1x1 refinement. Recall that the structural lemmas in Section 3 extend to orthostacks in general, and might lend insight to the more general problem. In particular, even with rectangles that are not rectangle-faced, if the faces are rectangular then a version of Lemma 2 applies, and there must be a pair of edges which at least partially border the two adjacent bands.

The obvious extension of our algorithm to orthostacks with only rectangular faces between the bands would be to again choose the bridge rectangle which has the highest x-coordinate, and unfold the adjacent band in the increasing x direction. However, in this case, since the bridge may not have an entire edge which is adjacent to

B_{i+1}, we lose the fact that we can arbitrarily glue rectangles as in step 1. This insight leads to an example of a rectangular orthostack (where all z-faces are rectangles but may have edges adjacent to each band) on which our algorithm will fail; see Figure 6. Note that in this example, the faces shaded red will overlap when unfolded via our algorithm.

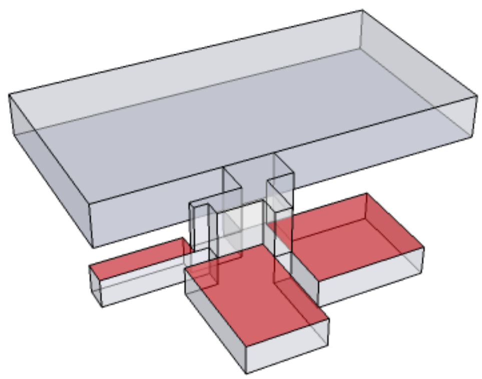

Figure 6: A rectangular orthostack (that is not rectangle-faced) where our algorithm fails to unfold into a planar polygon.

As an alternative, we propose the following algorithm. Instead of choosing the rectangle with the largest x-coordinate as our "bridge" between B_i and B_{i+1} in step 2, we could instead choose the rectangle which has the greatest width (measured so that this width has a component of B_i along one side and B_{i+1} along the opposite side, so that it could serve as a bridge). Intuitively, this rectangle separates the bands as much as possible, so that every other rectangle would have some point of attachment along the bands where it would fit without overlapping the neighboring band. The problem which remains is to show that none of the rectangles would overlap *each other*, since these rectangles are not rectangle-faced. This reduces to almost a type of matching argument; each rectangle has several possible attachment points, and we must find a selection so that no two overlap. It seems likely that Lemma 3 may prove useful here, since it provides a strong ordering on where the faces can be attached.

Extending this type of algorithm to non-rectangular orthostacks seems more difficult, since the notion of a good bridge would necessarily be more complex when z-faces are not simple rectangles. Choosing such a bridge would involve search for all possible ways that the z-faces could attach to the bands, and somehow finding the best (either "highest" or "widest") such bridge, as well as dealing with more complex overlap between z-faces when attached to the bands.

References

[1] B. Aronov and J. O'Rourke. Nonoverlap of the star unfolding. *Discrete Comput. Geom.*, 8:219–250, 1992.

[2] T. Biedl, E. D. Demaine, M. L. Demaine, A. Lubiw, J. O'Rourke, M. Overmars, S. Robbins, and S. Whitesides. Unfolding some classes of orthogonal polyhedra. In *Proc. 10th Canad. Conf. Comput. Geom.*, pages 70–71, 1998.

[3] M. Damian, R. Flatland, H. Meijer, and J. O'Rourke. Unfolding well-separated orthotrees. In *15th Annu. Fall Workshop Comput. Geom.*, pages 23–25, Nov. 2005.

[4] M. Damian, R. Flatland, and J. O'Rourke. Grid vertex-unfolding orthogonal polyhedra. In *Proc. 23rd Sympos. Theoret. Aspects Comput. Sci.*, volume 3884 of *Lecture Notes Comput. Sci.*, pages 264–276. Springer-Verlag, 2006.

[5] M. Damian, R. Flatland, and J. O'Rourke. Epsilon-unfolding orthogonal polyhedra. *Graphs and Combinatorics*, 23[Suppl]:179–194, 2007. Akiyama-Chvátal Festschrift.

[6] E. Demaine, J. Iacono, and S. Langerman. Grid vertex-unfolding orthostacks. In J. Akiyama, M. Kano, and X. Tan, editors, *Discrete and Computational Geometry*, volume 3742 of *Lecture Notes in Computer Science*, pages 76–82. Springer Berlin / Heidelberg, 2005.

[7] E. D. Demaine, D. Eppstein, J. Erickson, G. W. Hart, and J. O'Rourke. Vertex-unfoldings of simplicial manifolds. In A. Bezdek, editor, *Discrete Geometry*, pages 215–228. Marcel Dekker, 2003.

[8] E. D. Demaine and J. O'Rourke. *Geometric Folding Algorithms: Linkages, Origami, Polyhedra.* Cambridge University Press, 2007.

[9] J. Itoh, J. O'Rourke, and C. Vîlcu. Star unfolding convex polyhedra via quasigeodesic loops. *Discrete Comput. Geom.*, 44:35–54, 2010.

[10] J. O'Rourke. Unfolding orthogonal terrains. Technical Report 084, Smith College, July 2007. arXiv:0707.0610v4 [cs.CG].

[11] J. O'Rourke. Unfolding orthogonal polyhedra. In J. Goodman, J. Pach, and R. Pollack, editors, *Proc. Snowbird Conference Discrete and Computational Geometry: Twenty Years Later*, pages 307–317. American Mathematical Society, 2008.