

Grid Vertex-Unfolding Orthostacks

Erik D. Demaine*

John Iacono[†]

Stefan Langerman[‡]

Abstract

An algorithm was presented in [1] for unfolding orthostacks into one piece without overlap by using arbitrary cuts along the surface. It was conjectured that orthostacks could be unfolded using cuts that lie in a plane orthogonal to a coordinate axis and containing a vertex of the orthostack. We prove the existence of a vertex-unfolding using only such cuts.

1 Introduction

A long-standing open question is whether every convex polyhedron can be *edge unfolded*—cut along some of its edges and unfolded into a single planar piece without overlap [3]. A related open question asks whether every polyhedron without boundary (not necessarily convex but forming a closed orientable surface) can be *generally unfolded*—cut along its surface (not just along edges) and unfolded into a single planar piece without overlap. Biedl et al. [1] made partial progress on both of these problems in the context of *orthostacks*, i.e., orthogonal polyhedra for which every horizontal planar slice is a connected piece. They showed that not all orthostacks can be edge unfolded, but that all orthostacks can be generally unfolded. In their general unfoldings, all cuts are parallel to coordinate axes, but many of the cuts do not lie in coordinate planes that contain polyhedron vertices. Given the lack of pure edge unfoldings, the closest analog we can hope for with (nonconvex) orthostacks is to find *grid unfoldings* in which every cut is in a coordinate plane that contains a polyhedron vertex. In other words, a grid unfolding is an edge unfolding of the refined (“gridded”) polyhedron in which we slice along every coordinate plane containing a polyhedron vertex. Biedl et al. [1] asked whether all orthostacks can be grid unfolded.

We make partial progress on this problem by show-

*MIT Computer Science and Artificial Intelligence Laboratory, 32 Vassar St., Cambridge, MA 02139, USA, edemaine@mit.edu. Supported in part by NSF CAREER award CCF-0347776.

[†]Department of Computer and Information Science, Polytechnic University, 5 MetroTech Center, Brooklyn, NY 11201, USA, jiacono@poly.edu.

[‡]Chercheur qualifié du FNRS, Université Libre de Bruxelles, Département d’informatique, ULB CP212, Belgium. Stefan.Langerman@ulb.ac.be.

ing that every orthostack can be *grid vertex-unfolded*, i.e., cut along some of the grid lines and unfolded into a vertex-connected planar piece without overlap. Vertex-unfoldings were introduced in [2]; the difference is that faces can remain connected along single points (vertices) instead of having to be connected along whole edges. We also believe that our techniques can be extended to prove that all orthostacks can be grid unfolded, which would settle the open question of [1].

2 Unfolding algorithm

Given an orthostack K , let z_0, \dots, z_n be the distinct z coordinates of vertices of K . Subdivide the faces of K by cutting along every plane perpendicular to a coordinate axis that passes through a vertex of K . This subdivision *rectangulates* K . We use *rectangle* to refer to one element of this facial subdivision, while *face* refers to a maximal connected set of coplanar rectangles. We use *up* and *down* to refer to the z dimension, and use *left* and *right* to refer to the x dimension.

We partition the rectangles of K into several categories. After this categorization, the description of the unfolding layout is not difficult.

For $i = 0, 1, \dots, n - 1$, define the *i -band* to be the vertical rectangles (i.e., that lie in the xz plane or in the yz plane) whose z coordinates are between z_i and z_{i+1} . Each i -band is connected, and all of the rectangles of an i -band have the same extent in the z dimension, namely, $[z_i, z_{i+1}]$.

For $i = 0, 1, \dots, n$, we define the *i -faces* to be the faces of K in the horizontal plane $z = z_i$. An i -face has several properties. It may have the interior of K above or below it (but not both). For $0 < i < n$, its perimeter intersects with the $(i - 1)$ -band and the i -band. More precisely, the perimeter of the i -face has a nonempty intersection with the $(i - 1)$ -band, provided $i > 0$, and with the i -band, provided $i < n$. Each intersection is connected; i.e., the perimeter may be cut into two parts such that each part intersects solely the $(i - 1)$ -band or the i -band.

Also needed are the notions of the “begin rectangle” and “end rectangle” of the i -band. Choose the *0-band begin rectangle* to be an arbitrary rectangle on the 0-band. For $i \geq 0$, define the *i -band end rectangle* to be the rectangle of the i -band that is adjacent to the i -band begin

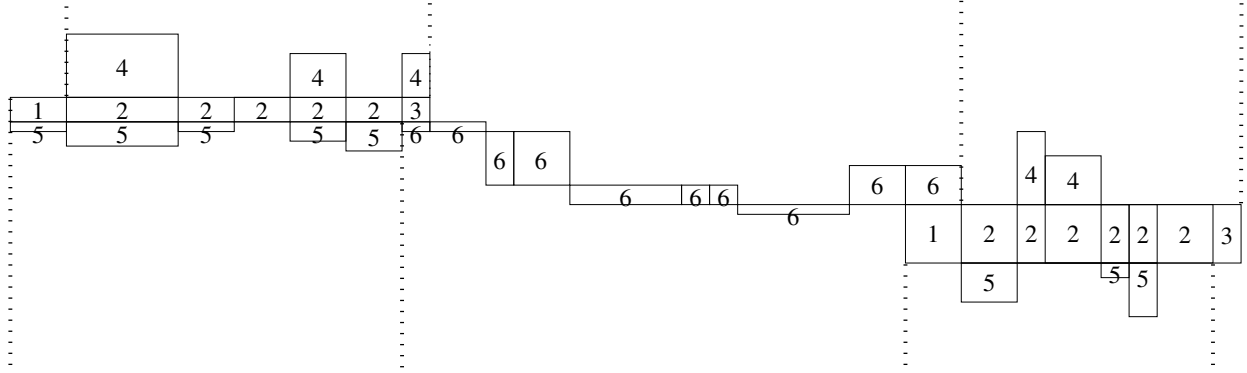


Figure 1: A hypothetical unfolded state produced by our algorithm. Rectangles are numbered according to their type: (1) i -band begin, (2) i -band interior, (3) i -band end, (4) $(i + 1)$ -down über, (5) i -up über, and (6) connecting.

rectangle in the clockwise direction as viewed from $+z$. (Thus, the begin and end rectangles of the i -band are adjacent.) For $i \geq 1$, define the i -connecting face to be the i -face that shares an edge with the $(i - 1)$ -band end rectangle. For $i \geq 1$, define the i -band begin rectangle to be one of the rectangles of the i -band that shares an edge with the i -connecting face. The i -band interior rectangles are rectangles of the i -band that are neither the begin rectangle nor the end rectangle.

Define the i -connecting set to be an edge-connected sequence of rectangles in the i -connecting face, starting at the rectangle that shares an edge with the $(i - 1)$ -band end rectangle and ending at the rectangle that shares an edge with the i -band begin rectangle. This sequence is chosen to contain the fewest rectangles possible (a shortest path in a dual graph on the rectangles in the i -connecting face), in order to preventing the path from looping around an island and thereby isolating interior portions of the i -face. Normal rectangles are rectangles of an i -face that are not in the i -connecting set.

We now unify all normal rectangles with their normal neighbors in the x dimension. Call the resultant rectangular regions *über-rectangles*. Thus i -faces are partitioned into the i -connecting rectangles and the i -über-rectangles. All i -über-rectangles are connected to the perimeter of the i -face, and thus are edge-connected to the i -band or the $(i + 1)$ -band. Define an i -up-über-rectangle to be an über-rectangle that is incident to the $(i + 1)$ -band and an i -down-über-rectangle to be a unified rectangle that is incident to the i -band. If an über-rectangle is incident to both, we classify it arbitrarily.

So, we have partitioned K into i -band begin rectangles, i -band end rectangles, i -band interior rectangles, i -up-über-rectangles, i -down-über-rectangles, and i -connecting rectangles. We now proceed with an informal description of the unfolding. Because of space constraints, a rigorous proof that the unfolding is non-

overlapping is omitted.

The unfolding has two components: the main component and the connecting component. For a given level i , the i -main component contains the i -band begin rectangle, the i -band end rectangle, the i -band interior rectangles, the $(i + 1)$ -down-über-rectangles, and the i -up-über-rectangles. The main component is laid out by cutting the edge between the i -band begin rectangle and the i -band end rectangle, and between all of the edges between the über-rectangles involved and (über-)rectangles not on the i -band. The layout induced by this edge-unfolding the remaining edges has, as the leftmost element, with nothing above, the i -band begin rectangle and as the rightmost element, with nothing below, the i -band-end rectangle.

The i -connecting component is used to connect the i -main component and the $(i + 1)$ -main component, using the i -connecting rectangles to connect the i -band end-rectangle at the right with the $(i + 1)$ -band begin rectangle at the left. We use the fact that an edge-connected sequence of rectangles can always be vertex unfolded so that each rectangle in the sequence lies to the left of the previous rectangle.

Figure 1 illustrates a hypothetical example of our unfolding.

References

- [1] T. Biedl, E. Demaine, M. Demaine, A. Lubiw, M. Overmars, J. O'Rourke, S. Robbins, and S. Whitesides. Unfolding some classes of orthogonal polyhedra. In *Proc. 10th Canad. Conf. Comput. Geom.*, Montréal, Aug. 1998.
- [2] E. D. Demaine, D. Eppstein, J. Erickson, G. W. Hart, and J. O'Rourke. Vertex-unfolding of simplicial manifolds. In *Proc. 18th ACM Symp. Comput. Geom.*, pp. 237–243, Barcelona, June 2002.
- [3] J. O'Rourke. Folding and unfolding in computational geometry. In *Revised Papers from the Japan Conf. Discrete Comput. Geom.*, LNCS 1763, pp. 258–266, Tokyo, Dec. 1998.