Pacific Graphics (2011)
Bing-Yu Chen, Jan Kautz, Tong-Yee Lee, and Ming C. Lin (Editors)

Interactive Design of Planar Curved Folding by Reflection

J. Mitani[1,3] and T. Igarashi[2,3]

[1]University of Tsukuba, Japan
[2]The University of Tokyo, Japan
[3]JST ERATO

Abstract

Designing curved folds of a developable surface remains an open problem. No analytical method for the design of curved folds has been reported. Although optimization methods exist, it is difficult to interactively design valid curved folds of a single developable surface using such methods. In the present paper, we propose an interactive system for designing a surface with planar curved folds based on mirror reflection. The key consideration is that a surface created by sequential reflective folding is guaranteed to be isometrically unfolded into a plane. We demonstrate that the proposed system enables the interactive design of a variety of expressive curved folds, which was very difficult using previous methods.

Categories and Subject Descriptors (according to ACM CCS): I.3.5 [Computer Graphics]: Computational Geometry and Object Modeling—Curve, surface, solid, and object representations

1. Introduction

Developable surfaces have valuable characteristics for industrial and architectural design, including reconstructability with a flat thin material without tearing or stretching. Although some methods for designing a surface by gluing multiple developable sheets have been proposed [MS04, STL06, RSW*07], it is still difficult to design a surface with curved folds that is reconstructable with a single developable sheet without cutting or gluing.

Designing a shape by folding a single sheet of paper has been investigated extensively in the field of origami. We herein refer to such shapes as *origami surfaces*. Mathematically, an origami surface is a surface that consists of multiple developable patches and can be isometrically unfolded into a plane without cutting along creases. This type of surface is sometimes referred to as an *applicable surface* [KGK94].

Although folding problems have been investigated extensively, the method by which to design a shape with curved folds remains a difficult problem. Kilian et al. [KFC*08] recently proposed a method for digitally reconstructing the geometry of curved folds from scanned 3D data using an optimization-based approach. They established a basis for discussing curved folds using discrete representations.

We admit handling curved folds is still a difficult problem. However, by limiting the curve to be planar, the problem becomes drastically simple. In the present paper, we propose a system in which the user can interactively design an origami surface with planar curved folds. Kilian et al. [KFC*08] also proposed a method for designing curved folds with which the user draws a curve on the surface. The folded shape is generated by using the optimization. On the other hand, our approach is based on the mirror reflection, and the folded shape is analytically obtained. Therefore, our approach is accurate and fast under the limitation that the curve is planar. The user grabs and drags a point on a surface, and the system continuously displays the folded result. The key idea is to use mirror reflection through an implicitly defined plane. Beginning from a simple initial origami surface, the user can explore complicated origami surfaces with multiple curved folds by continuously adding folds.

1.1. Theoretical basis

It is known that when the curved fold is contained in a single plane, the pair of generating lines at any point on the fold makes equal angles with that plane [Huf76]. This means that the plane acts as a reflection plane that reflects the tangent plane on one side to the tangent plane on the other side. This implies the closure property of the origami surface under mirror reflection when we do not take account of self-intersections, which can be expressed as the following theorem:

Theorem (closure property): *A surface generated by applying mirror reflection to a part of an origami surface is also an origami surface.*

Proof: When a plane that crosses an origami surface S divides this surface into surfaces s_1 and s_2, the reflection s_2' of s_2 through the plane is also an origami surface. Here, s_2' connects to s_1 without any gaps at the cross section of S and the plane (Figure 1). The development of s_2' also connects to the development of s_1 without any gaps because the shape of the development of s_2 does not change by reflection. ■

This fact indicates that a complicated origami surface with planar curved folds is obtained by applying multiple reflections to a simple origami surface.

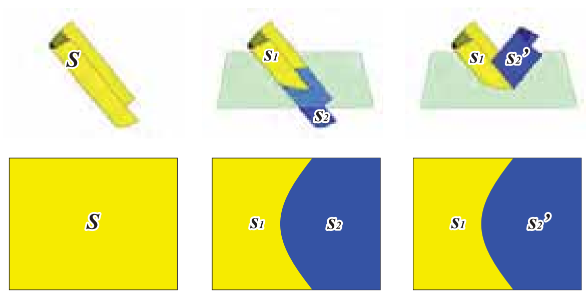


Figure 1: *A shape generated by applying mirror reflection to part of an origami surface is also an origami surface. Left: Initial origami surface S and the development. Middle: A plane intersecting S divides the surface into surfaces s_1 and s_2. Right: s_2 is reflected through the plane to s_2'.*

1.2. Flow of the design process

With the proposed system, the editing starts from an initial origami surface represented by triangle mesh. The surface will be a simple developable surface without any folds such as a part of a cone or a cylinder. The model has two states, *fixed* (Figure 2(a, d)) and *temporal* (Figure 2(b, c, e, f)), during editing operations. Initially the model is in the *fixed* state. When the user selects a grabbing point on the surface, the state changes to *temporal*. Along with the transfer of the point, the surface is updated until the user changes the state to *fixed*. The flow of the design process is as follows (Figure 2):

1. Prepare an initial origami surface.
2. Select a vertex on the surface.
3. Move a vertex by dragging a handle associated with the vertex. The shape is automatically updated while maintaining the constraint that the surface is an origami surface.
4. Fix the position of the selected vertex.
5. Repeat step 2 through 4 until the user satisfies.

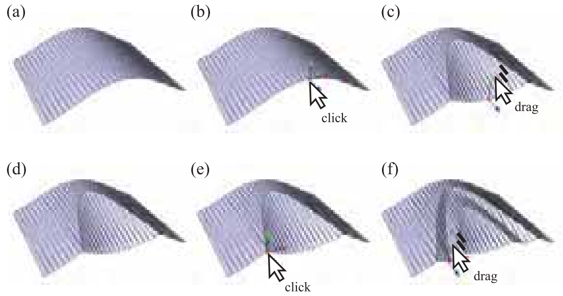


Figure 2: *Flow of the design process. (a) Input an initial origami surface. (b, e) Select a vertex on the surface. (c, f) Drag and move the vertex. The surface is automatically modified so that the surface maintains developability by adding curved folds. (d) Fix the position of the selected vertex.*

1.3. Contribution

In this paper, we present an interactive design system that allows a user to create a wide range of complicated origami surfaces with multiple curved folds. Although a number of curved origami have already been designed manually by applying reflections to a conical surface [DDK10], no interactive system has yet been presented. We expect the proposed system to be used as a creative and explorative tool for designers.

2. Related Work

There have few studies on the analytical computation of curved folds. Kergosien et al. [KGK94] and Frey [Fre04] computed the creases generated when a developable surface buckles. Although appropriate shapes are calculated, the intended curved folds are not designed. Mitani [Mit09] proposed a method for designing axisymmetric curved origami, which consists of cylindrical surfaces with curved folds. A subset of his origami, which consists of planar curved folds, is reconstructable using the proposed approach.

David Huffman's origami sculptures [WER04] are well-known masterpieces designed using curved folds. Demain et al. [DDK10] tried to reconstruct Huffman's origami in order to investigate the principles used in the design. They clarified that some pieces are designed using reflection to a truncated cone. However, the method by which to design general curved folds remains unknown.

Fuchs and Tabachnikov [FT99] described that "if the fold is a nonclosed arc, the folded paper tends to occupy such a position that the ridge lies in a plane". This means that a planar curve is a physically stable state of a curved fold. Although when the curvature of the fold is large, as our experimental observation, the fold tends to slightly open according to elasticity of paper, the description is almost true. Based on this view, the limitation of the proposed approach, in which generated folds are only planar curves, would be appropriate for designing physically stable curved folds.

Few studies have examined the interactive design of curved folds. Although interactive methods for designing or simulating curved developable surfaces have been proposed [BW07,EB08], these methods do not consider folding or adding creases, but only bending. Kilian et al. [KFC*08] provided a simple interface for adding a new fold to a surface by drawing a curve on the surface. The shape folded along the curve is generated by applying an optimization. Since the user sees the folded shape only after the optimization, it is difficult to quickly attempt several folds.

3. Proposed System

3.1. Initial surface

The user first provides an initial origami surface to the system. The proposed system does not check the developability of the initial surface, and so it must be ensured by the user. A simple developable patch, such as a cylindrical or conical surface is usable. It is also possible to design an initial surface using an existing geometric modeler, e.g., sweeping a planar curve along a straight line generates a developable surface. The initial surface can have holes. The topology of the resulted shape is always same with the initial surface.

3.2. User interface

A new fold is added by reflecting a part of the origami surface through a plane. A simple approach would be as follows.

The user specifies the location and orientation of the plane of reflection. The user then specifies which part is to be reflected (because there are multiple choices, as shown in Figure 3).

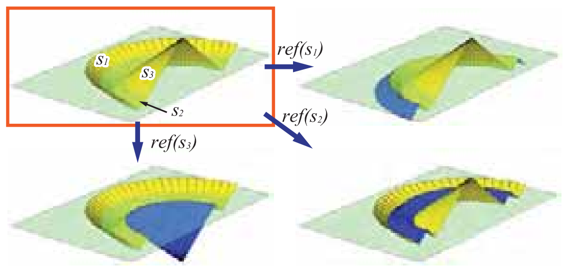


Figure 3: *When a refection plane divides the surface into three regions s_1, s_2, and s_3 , there are three choices for reflection. $ref(s_i)$ denotes a reflection operation of s_i.*

However, providing an intuitive interface for specifying the position and location of the reflection plane is not straightforward. Furthermore, the user cannot see the folded shape until the user completes the operation.

We therefore implemented the following user interface in order to achieve a more intuitive and interactive interface, with which the user can edit an origami surface continuously as described in Section 1.2.

The user selects a vertex on the origami surface. The user then moves the vertex by dragging a handle associated with the vertex. The folded shape is automatically updated as the vertex moves while maintaining the condition that the shape is an origami surface.

3.3. Algorithm

We represent an origami surface as a discrete surface by using a triangular mesh for easy implementation. Planar quadrilaterals are often used for the discrete representation of developable surfaces in order to ensure developability [LPW*06, KFC*08]. However, a triangular mesh can be applied without a problem in the proposed system because the generated shape is guaranteed to be an origami surface when the initial surface is an origami surface. In the proposed system, every triangle contains references to three vertices. Every vertex also contains references to triangles which contain the vertex iteself in order to make it possible to obtain triangles which share a common vertex in constant time.

The plane of reflection is implicitly defined as a bisection plane between the initial and current positions of the selected vertex (Figure 4). The system divides the triangle mesh into multiple parts by the plane and reflects only the part that contains the selected vertex. With this user interface, the user can select and move a point while seeing the shape being continuously transformed.

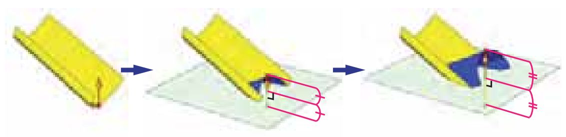


Figure 4: *When the user drags a vertex (indicated by a red point) (left), the system reflects the surface at the bisection plane between the initial and current positions of the vertex (middle and right). The reflected part shown in blue is defined as the part that contains the selected vertex.*

The system internally holds two triangle meshes, M_{fixed} and $M_{temporal}$. M_{fixed} represents a fixed state of an origami surface. At the beginning, M_{fixed} is the initial origami surface. When the user selects a vertex on M_{fixed}, the system copies M_{fixed} to $M_{temporal}$. $M_{temporal}$ represents the temporal state of an origami surface which is displayed on a screen while the user moves the selected vertex. Each time the location of the selected vertex is changed, $M_{temporal}$ is updated by reflecting a part of M_{fixed} through the reflection plane. After the move of the selected vertex is finished by the user (e.g. by pressing a certain key), the current $M_{temporal}$

is copied to M_{fixed}. Through this sequence of operations, a fold is added to the surface. This is repeated until the user satisfies with the result.

When the reflection plane divides the mesh into multiple parts as shown in Figure 3, the part to be reflected is only a part which contains the selected vertex. The triangles that do not belong to the part are not divided even if the reflection plane crosses. Whether a triangle belongs to the reflected part or not is distinguished by recursively visiting neighboring triangles by referring the vertex-triangle references starting from the selected vertex. The triangles only a portion of which belong to the reflected part are divided into sub-triangles by the reflection plane. On the other hand, the triangles that completely belong to the reflected part are reflected through the reflection plane without division. This means, for example, when no triangles cross the reflection plane, the entire surface is mirrored through the plane. The system does not check for self collisions. As such, the user can temporarily create a shape with self-intersections on the way to the final goal.

4. Results

Example origami surfaces newly designed using the proposed system by the author are shown in Figures 5(a) through 5(e). As a user study, we asked a computer graphics designer to freely design a shape. Figure 5(f) shows the result. Each resulting shape (middle image of each row) was obtained by applying at most 10 reflections to initial shape (left image of each row). A few minutes were required to interactively design each shape. Each development of an origami surface (right image of each row) was generated by simply unfolding triangles on a plane. Only sharp edges were shown on the development. The system implemented on a standard PC (Intel Core2 Duo CPU T9900 3.06GHz, 4GB RAM) was quick enough to edit interactively all of the origami surfaces shown in Figures 5. Real models made from a sheet of paper folded by referring to the developments are shown in Figure 6. We used glue to close a slit in the model generated from a cone (Figure 5(f)). Further, we fixed some points of the models to make it stable.

We interviewed the designer after the design to obtain subjective comments. The designer reported that he enjoyed casually discovering artistic forms and that he was impressed by the ability to confirm that the shape can actually be constructed by simply folding a single sheet of paper. The time required for design (Figure 5(f)) was only a few minutes. On the other hand, the designer also noted that it was difficult to design intended shapes. Determining whether it is possible to create an imagined shape with an origami surface appears to be difficult.

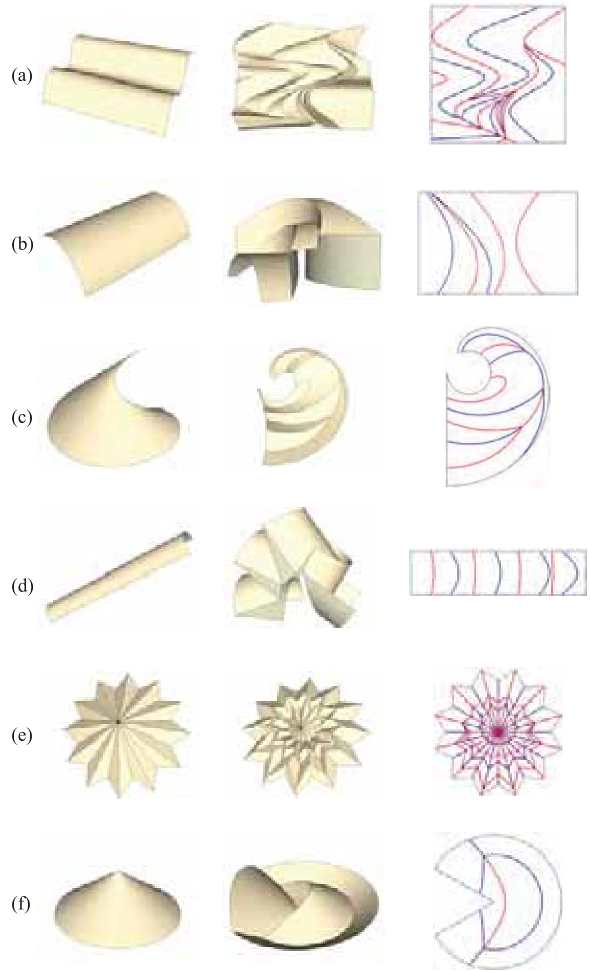
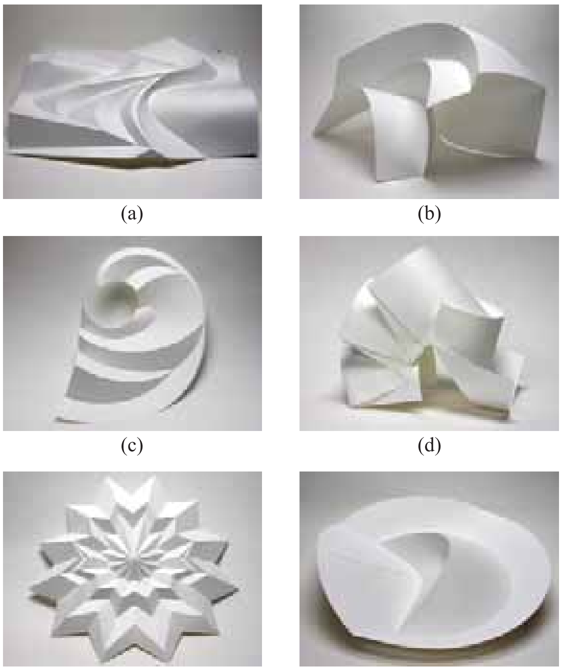


Figure 5: *From left to right: initial surface, resulting shape, and unfolded pattern. (Red and blue lines correspond to mountain and valley folds, respectively.) The initial surfaces are (a, b) a curved surface generated by sweeping a plane curve, (c) a tangent developable surface of a helix, (d) half of cylinder, (e) a star that is flattened without cuts, and (f) a cone (with a slit).*

5. Limitations and Future Work

Reflection generates only planar folds. Therefore, the user cannot create an origami surface with non-planar folds as introduced in the literature [WER04, KFC*08, KDD08] (Figure 7).

In the future, we intend to explore a more general principle for interactively generating non-planar curved folds. The design process in the proposed system is rather explorative. The user can quickly explore various folds before eventually reaching an aesthetically pleasing result. This is different from the typical modeling procedure in which the user starts



(a) (b) (c) (d) (e) (f)

Figure 6: *Photographs of folded paper designed using the proposed prototype system.*

Figure 7: *Non-planar curved folds introduced in [KDD08, WER04]. Reconstructed by the author.*

with a given goal shape and then determines a sequence of folds to achieve the goal. This is inevitable because designing an intended shape with a developable surface is essentially a highly constrained design problem. However, we hope to develop a method for supporting a more goal-oriented modeling process in the future. An important step toward accomplishing this goal is to find solutions to open problems associated with reflective folding. For example, is any origami surface in which all folds are planer reconstructable by applying sequential mirror reflections to a single developable surface?

We use a 2D input device (mouse) as an input device, but this is not an optimal user interface for 3D folds. A two-handed user interface [HPGK94] would be applicable in which the non-dominant hand with 6-DOF tracking holds the base sheet and the dominant hand with 3-DOF tracking holds a point to be reflected. We expect that this setting will further accelerate the design process.

References

[BW07] Bo P., Wang W.: Geodesic-controlled developable surfaces for modeling paper bending. *Computer Graphics Forum 26* (2007), 365–374. 3

[DDK10] Demaine E., Demaine M., Koschitz D.: Reconstructing david huffman's legacy in curved-crease folding. In *Origami5: Proceedings of the 5th International Conference on Origami in Science, Mathematics and Education* (2010), p. AD12. 2

[EB08] English E., Bridson R.: Animating developable surfaces using nonconforming elements. *ACM Trans. Graph. 27* (August 2008), 66:1–66:5. 3

[Fre04] Frey W.: Modeling buckled developable surfaces by triangulation. *Computer Aided Design 36* (2004), 299–313. 2

[FT99] Fuchs D., Tabachnikov S.: More on paperfolding. *The American Mathematical Monthly 106* (January 1999), 27–35. 2

[HPGK94] Hinckley K., Pausch R., Goble J. C., Kassell N. F.: Passive real-world interface props for neurosurgical visualization. In *Proceedings of the SIGCHI conference on Human factors in computing systems: celebrating interdependence* (New York, NY, USA, 1994), CHI '94, ACM, pp. 452–458. 5

[Huf76] Huffman D. A.: Curvature and creases: A primer on paper. *IEEE Trans. Comput. 25* (October 1976), 1010–1019. 1

[KDD08] Koschitz D., Demaine E., Demaine M.: Curved crease origami. In *Abstracts from Advances in Architectural Geometry* (2008), pp. 29–32. 4, 5

[KFC*08] Kilian M., Flöry S., Chen Z., Mitra N. J., Sheffer A., Pottmann H.: Curved folding. *ACM Trans. Graph. 27* (August 2008), 75:1–75:9. 1, 3, 4

[KGK94] Kergosien Y. L., Gotoda H., Kunii T. L.: Bending and creasing virtual paper. *IEEE Comput. Graph. Appl. 14* (January 1994), 40–48. 1, 2

[LPW*06] Liu Y., Pottmann H., Wallner J., Yang Y.-L., Wang W.: Geometric modeling with conical meshes and developable surfaces. *ACM Trans. Graph. 25* (July 2006), 681–689. 3

[Mit09] Mitani J.: A design method for 3d origami based on rotational sweep. *Computer-Aided Design and Applications 6* (2009), 69–79. 2

[MS04] Mitani J., Suzuki H.: Making papercraft toys from meshes using strip-based approximate unfolding. *ACM Trans. Graph. 23* (August 2004), 259–263. 1

[RSW*07] Rose K., Sheffer A., Wither J., Cani M.-P., Thibert B.: Developable surfaces from arbitrary sketched boundaries. In *Proceedings of the fifth Eurographics symposium on Geometry processing* (Aire-la-Ville, Switzerland, 2007), Eurographics Association, pp. 163–172. 1

[STL06] Shatz I., Tal A., Leifman G.: Paper craft models from meshes. *Vis. Comput. 22* (September 2006), 825–834. 1

[WER04] Wertheim M.: Cones, curves, shells, towers: He made paper jump to life. The New York Times, June 22 2004. 2, 4, 5