Journal of Computational Design and Engineering 5 (2018) 442–448

Contents lists available at ScienceDirect

Journal of Computational Design and Engineering

journal homepage: www.elsevier.com/locate/jcde

Approximating 3D surfaces using generalized waterbomb tessellations

Yan Zhao *, Yuki Endo, Yoshihiro Kanamori, Jun Mitani

University of Tsukuba, 1-1-1 Tennoh-dai, Ibaraki 305-0006, Japan

ARTICLE INFO

Article history:
Received 7 April 2017
Received in revised form 22 November 2017
Accepted 7 January 2018
Available online 9 January 2018

2010 MSC:
00-01
99-00

Keywords:
Computational origami
Approximation
Developable constraint
Computer aided design

ABSTRACT

Origami has received much attention in geometry, mathematics, and engineering due to its potential to construct 3D developable shapes from designed crease patterns on a flat sheet. Waterbomb tessellation, which is one type of traditional origami consisting of a set of waterbomb bases, has been used to create geometrically appealing 3D shapes and been widely studied. In this paper, we propose a method for approximating target surfaces, which are parametric surfaces of varying or constant curvatures, using generalized waterbomb tessellations. First, we generate a base mesh by tiling the target surface using waterbomb bases. Then, by applying a simple numerical optimization algorithm to the base mesh, we achieve a developable waterbomb tessellation, which can be developed onto a plane without stretching. We provide a prototype system using which the user can adjust the resolution of the tessellation and modify waterbomb bases. Our work could expand the exploration of building developable 3D structures using origami.

1. Introduction

Origami, also known as paper folding, has the potential to construct 3D shapes by folding thin sheets of paper along predefined creases without introducing cuts and distortions. An origami piece can be defined by its crease pattern, which contains a set of mountain and valley folded lines (shown in red and blue in this paper) appearing on a sheet of paper when the origami is opened flat. The crease pattern is scale independent and thus can be applied at the nanometric (Edwards & Yan, 2014; Nangreave, Han, Liu, & Yan, 2010; Rothemund, 2006; Tørring, Voigt, Nangreave, Yan, & Gothelf, 2011) or space level (Lang, 2009; Miura, 1989; Pohl & Wolpert, 2009; Wilson, Pellegrino, & Danner, 2013).

Among the types of origami, waterbomb tessellation is a traditional one that can be used to create geometrically appealing 3D shapes, such as that in Fig. 1(a). As shown in Fig. 1, a 3D waterbomb origami (a) is defined by its crease pattern (b), which contains a set of waterbomb bases (c). Such origami pieces are developable, which is guaranteed by the fact that the sum of the sector angles around each interior vertex equals 360°. The waterbomb base, which is also referred to as a regular base, has a mirror-symmetric property. The base has the geometric feature containing four valley and two mountain folded lines meeting at the center vertex. Here, we introduce a generalized waterbomb base (Fig. 1(d)) that inherits this geometric feature but could omits the mirror-symmetric property. Furthermore, we introduce a generalized waterbomb tessellation that contains generalized waterbomb bases to approximate target 3D surfaces.

Waterbomb tessellation has also been widely studied. Tachi, Masubuchi, and Iwamoto (2012) analyzed the kinematics of waterbomb tessellations to achieve adaptive freeform surfaces. They generated a model based on the multiple degree of freedom inherent in waterbomb tessellation containing regular bases. Additionally, Kuribayashi et al. (2006) made the first origami stent to achieve a large deployable ratio. Onal, Wood, and Rus (2013) demonstrated a worm robot, and Lee, Kim, Kim, Koh, and Cho (2013) fabricated a deformable wheel robot. Chen, Feng, Ma, Peng, and You (2016) proposed a comprehensive kinematic analysis on a waterbomb origami with one degree of freedom motion under symmetric folding.

In this paper, we approximate target surfaces, which are parametric surfaces of varying or constant curvatures, using generalized waterbomb tessellations. An overview of our method is shown in Fig. 2. We take a 3D parametric surface, e.g., Fig. 2(a), as input. Then, we sample u and v coordinates in the parametric uv-plane to achieve a quad approximation (Fig. 2(b)). Next, we

Peer review under responsibility of Society for Computational Design and Engineering.

* Corresponding author.
E-mail addresses: yanzhao.npal.tsukuba@gmail.com (Y. Zhao), endo@cs.tsukuba.ac.jp (Y. Endo), kanamori@cs.tsukuba.ac.jp (Y. Kanamori), mitani@cs.tsukuba.ac.jp (J. Mitani).

https://doi.org/10.1016/j.jcde.2018.01.002
2288-4300/

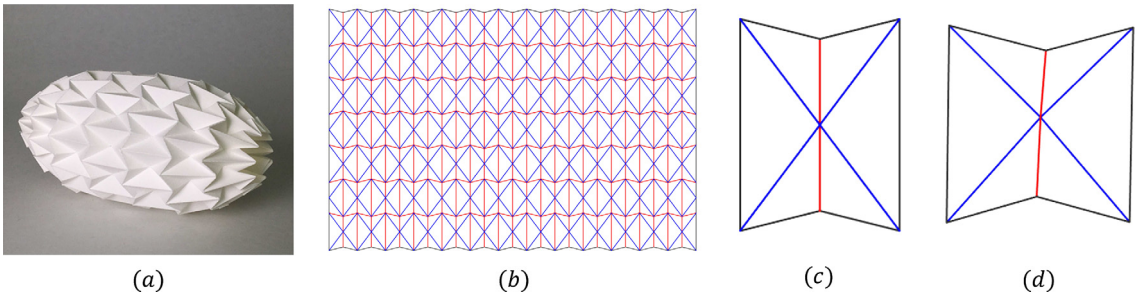

Fig. 1. Geometry of generalized waterbomb origami.

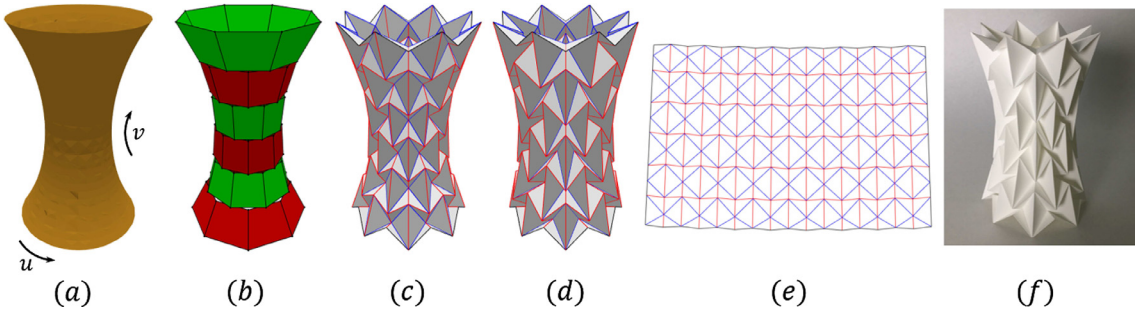

Fig. 2. An overview of our method.

generate a base mesh (Fig. 2(c)) by creating waterbomb bases in the quads. Here, our prototype system enables us to generate base meshes with variable resolutions and modify waterbomb bases interactively. Then, by applying a simple numerical optimization algorithm to the base mesh, we achieve a developable waterbomb tessellation (Fig. 2(d)), which can be developed onto a plane without stretching. Finally, the user can fold the crease pattern (Fig. 2(e)) to achieve the origami piece (Fig. 2(f)). We demonstrate several resulting approximations, which expands the exploration of building developable structures using origami.

The rest of this paper is organized as follows. Related work is discussed in Section 2. Section 3 presents a detailed description of our method. In Section 4, we demonstrate several developable approximations. We conclude this paper in Section 5 and discuss future work.

2. Related work

TreeMaker is software used to design flat-foldable origami (Lang, 2006). Its basic concept was first introduced by Meguro (1991) and fully described by Lang (1996). This software generates a crease pattern from a graph tree that represents the base structure of the object by using a circle/river packing technique. *Tess* is another computer program that makes crease patterns for origami tessellations and involves twist folds in a repeating pattern (Bateman). These approaches focus on flat-foldable origami, while we aim at approximating 3D surfaces using origami which may not be flat-foldable.

Mitani proposed methods for designing 3D origami on the basis of rotational sweep (Mitani, 2009, 2011). The methods generate a crease pattern for an axisymmetric structure by adding flaps outside of the target shape. Although the flaps might be considered obtrusive, his method succeeds in generating 3D curved origami. One of his methods (Mitani, 2012), which combines the advantages of the rotational sweep and mirror reflection approaches, has been used to build geometrically attractive origami pieces. Mitani & Igarashi (2011) also proposed an interactive system that allows the user to design 3D curved origami surfaces with mirror operations specified by selecting and moving vertices on the 3D origami while maintaining the developability of the resulting shape.

Zhao, Kanamori, and Mitani (2017) proposed a method for handling a family of axisymmetric 3D origami consisting of triangle facets. This method first designs a rotationally-symmetric crease pattern and then calculates an axisymmetric 3D origami piece on the basis of geometric constraints. By adding a cut in the crease pattern, such 3D origami can be axisymmetrically deployed or flat-folded by changing one parameter. More recently, Zhao, Kanamori, and Mitani (2018) proposed a method for designing axisymmetric 3D origami based on a mirror-symmetric crease pattern. The method explored the variations of the calculated 3D origami and presented a rigid folding motion that can axisymmetrically deploy or flatten the 3D shape.

For approximating 3D surfaces using origami, Tachi proposed the *Origamizer* algorithm (Tachi, 2009, 2010a), which generates crease patterns for arbitrary 3D triangle mesh models with a topological disc condition. Then, he proposed a system (Tachi, 2013) for approximating a target shape by using a subset of generalized Resch patterns. However, these approaches were based on the tucking technique, which hides unnecessary areas of a sheet of paper inside the shape. He also proposed a design system *Freeform Origami* (Tachi, 2010b), which allows the user to vary a known origami in 3D while preserving the developability and other optional conditions inherent in the crease pattern. Through dragging the vertices in 3D, the system enables the user to edit a given pattern into a freeform. However, the method cannot fully support approximating target 3D surfaces.

In addition, several approximating approaches based on modified Miura-ori have been proposed. Zhou, Wang, and You (2015) developed a vertex method for generating developable 3D origami between two singly curved surfaces. Song, Zhou, Zang, Wang, and You (2017) proposed a mathematical framework for the generation of rigid-foldable 3D origami based on the crease pattern that can simultaneously fit two doubly curved surfaces with rotational symmetry about a common axis. Dudte, Vouga, Tachi, and Mahadevan (2016) used modified Miura cells to approximate

orienTable 3D surfaces with positive, zero, negative, and mixed Gauss curvatures. In this paper, we focus on another basic origami tessellation, waterbomb tessellation, to fit onto target surfaces.

3. Approximating target surfaces

We demonstrate the generation of a base mesh in Section 3.1. Optimizing the base mesh to achieve a developable approximation is discussed in Section 3.2.

3.1. Generation of base mesh

The generation of the base meshes on parametric surfaces is versatile; we can generate base meshes on axisymmetric or non-axisymmetric target surfaces and on orientable or non-orientable target surfaces. We tile a given surface using quads for the initial approximation. Parametric surfaces are taken as input in this work. Therefore, we can easily achieve a set of quads by isometrically sampling u and v coordinates, which vary within a certain domain D in the parametric $uv-$plane, of the input parametric surface.

Hereafter, we explain the case of a catenoid surface as an example. A catenoid surface (Fig. 2(a)) is defined with $u,\ v$ parameters as:

$$P(x,y,z) = \left(\cosh\frac{v}{c}\cos u, \cosh\frac{v}{c}\sin u, v\right), \tag{1}$$

where $u \in [0, 2\pi],\ v \in [-\pi, \pi]$, and c is a non-zero real constant that is set as 2.5 in this case. As shown in Fig. 3(a) and (b), we isometrically sample u and v coordinates to achieve sampling points. The steps of u and v for sampling are denoted as Δu and Δv, which equal $2\pi/N_u$ and $2\pi/N_v$, respectively. N_u indicates the number of quads in one strip, which is shown in red and green. N_v means the number of strips used for constructing the approximation. Both N_u and N_v are integers and set as 10 in this case.

As can be observed from waterbomb tessellations, adjacent strips are shifted against each other by $\Delta u/2$ in the u direction in the uv-plane. A naïve way of doing this is to shift only odd strips by $-\Delta u/2$. However, this works with axisymmetric shapes but fails with non-axisymmetric ones because quads along boundaries become jagged and cannot cover the target surfaces. To handle both axisymmetric and non-axisymmetric shapes, we first temporarily generate $N_u + 1$ quads for odd strips. Suppose that parameter u in the given parameter surfaces ranges from u_s to u_e. Here, $N_u + 1$ quads are achieved in the range from $u_s - \Delta u/2$ to $u_e + \Delta u/2$. In particular, the first quad's u ranges from $u_s - \Delta u/2$ to $u_s + \Delta u/2$, and the last quad's u ranges from $u_e - \Delta u/2$ to $u_e + \Delta u/2$ (Fig. 3(b)). In the case of axisymmetric shapes, the first and the last quads are identical because parameter u is periodic. We then generate a waterbomb base in each quad and select only a half of the first and the last waterbomb bases to ensure N_u waterbomb bases in each strip (shown in Fig. 5 and discussed below).

During the initial approximation using quads, we allow the user to adjust the density of quads by changing N_u and N_v interactively. Fig. 3(c) is an approximation created by double density sampling both in the u and v directions, and thus, it has four times more quads than that in Fig. 3(a) to represent the target surface. The more quads we use, the more accurately we can approximate the target surface. Considering fabrication by paper-folding, however, we also have to consider the increase of labor. Balancing the approximation accuracy and fabrication labor is an interesting problem, which is left as future work.

Next, we generate each waterbomb base by adding and moving three auxiliary vertices, $P_u,\ P_d$, and P_c, as shown in Fig. 4. At an initial state (Fig. 4(a)), P_d and P_u are the midpoints of segments P_1P_2 and P_3P_4, respectively; P_c is the midpoint of segment P_dP_u. We connect boundary points $P_1,\ P_2,\ P_3,\ P_4,\ P_d$, and P_u to P_c. The positions of $P_1,\ P_2,\ P_3$, and P_4 are fixed. We then move $P_u,\ P_d$, and P_c to form a structure that "looks" like a waterbomb base, with four valley and two mountain folded lines. Specifically, P_c can be moved along the normal of quad $P_1P_2P_3P_4$ (Fig. 4(b)). P_u can be "dragged down" by rotating it in plane $P_uP_cP_d$ (Fig. 4(c)). Similarly, P_d can be "dragged up" by rotating it in plane $P_uP_cP_d$ (Fig. 4(d)).

Finally, we merge adjacent bases to achieve a base mesh (Fig. 5(c)). Fig. 5(a) is an approximation with gaps. By averaging the positions of adjacent vertices (b), we achieve a base mesh without gaps as shown in (c). Note that there are $N_u + 1$ bases for odd strips. Here, we select only the right part of the first base and the left part of the last base to ensure N_u bases in the odd strips.

3.2. Numerical optimization

In this section, we apply a simple numerical optimization to base meshes to produce developable surfaces. We use an angle constraint (Tachi, 2010b), which requires that the total angle around a developable vertex i be 360°:

$$D_i = 360^\circ - \sum_{k=0}^{K_i-1}\alpha_{i,k} = 0, \tag{2}$$

where K_i is the total number of sector angles around vertex i, and $\alpha_{i,k}$ is the k-th incident sector angle of vertex i. In our work, we classify vertices as interior vertices having six adjacent facets and boundary

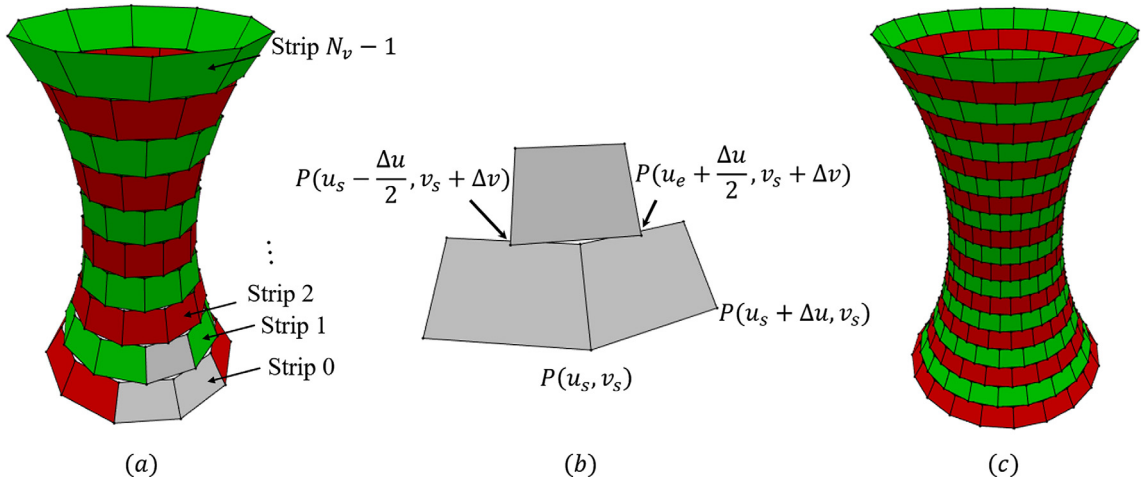

Fig. 3. Initial approximation using quads.

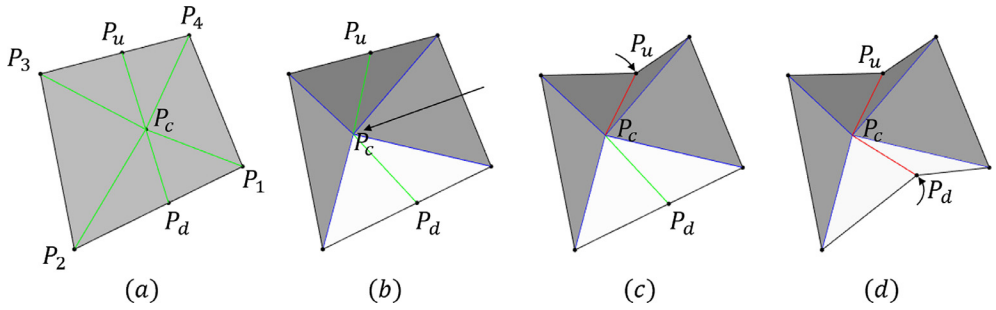

Fig. 4. Modification of waterbomb base

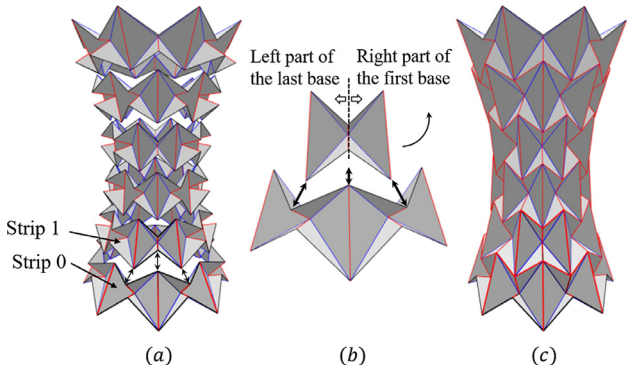

Fig. 5. Merging waterbomb bases to achieve base mesh.

vertices having less than six adjacent facets. For a developable surface, all the interior vertices should satisfy the angle constraint.

We implemented the Levenberg-Marquardt algorithm to solve such an optimization problem. Boundary/interior vertices are viewed as fixed/free nodes, respectively. For each iteration of the Levenberg-Marquardt algorithm, we evaluate the maximum α_{max}, minimum α_{min}, and average α_{ave} of the sum of angles around each interior vertex for a termination criterion. Correspondingly, we introduce the errors of e_{max}, e_{min}, and e_{ave} represented as:

$$\begin{aligned} e_{max} &= |360^\circ - \alpha_{max}|. \\ e_{min} &= |360^\circ - \alpha_{min}|. \\ e_{ave} &= |360^\circ - \alpha_{ave}|. \end{aligned} \tag{3}$$

Moreover, we introduce e_{total} as the maximum among e_{max}, e_{min}, and e_{ave}. The procedure is terminated when e_{total} is less than e_d. In our experiment, we set e_d as $1e{-}5$ to produce a developable surface.

Fig. 6 shows graphs of convergence created during optimization on the base mesh (Fig. 2(c)), where Fig. 6(a) shows the relationship between the numbers of iterations and values of α_{max}, α_{min}, and α_{ave}. Correspondingly, Fig. 6 (b) demonstrates the values of e_{max}, e_{min}, and e_{ave} that calculated during the iterations. In this case, e_{total} becomes less than $1e{-}5$ when the number of iterations is 198.

4. Results

We developed a prototype system using Java to implement our method. We ran our system on an Intel(R) Core(TM) i7-4770 CPU with an 8-GB-RAM PC. For a given target surface, our method allows the user to generate base meshes with variable resolutions and then produces developable approximations. As shown in Fig. 7, we show four results, each of which contains a base mesh, its corresponding approximation, and the approximation with the target surface, as shown in Fig. 2(a).

Table 1 shows the parameters in detail and the results of the models shown in Fig. 7. e_{total} of each approximation was less than $1e{-}5$ after optimization, with which we consider the approximation to be developable. To evaluate the difference between the resultant approximation A and the target surface T, we define distance $d(A, T)$ as:

$$\begin{aligned} d(A,T) &= \text{mean}[d(x,T)], x \in A, \\ d(x,T) &= \min[d(x,y)], y \in T, \end{aligned} \tag{4}$$

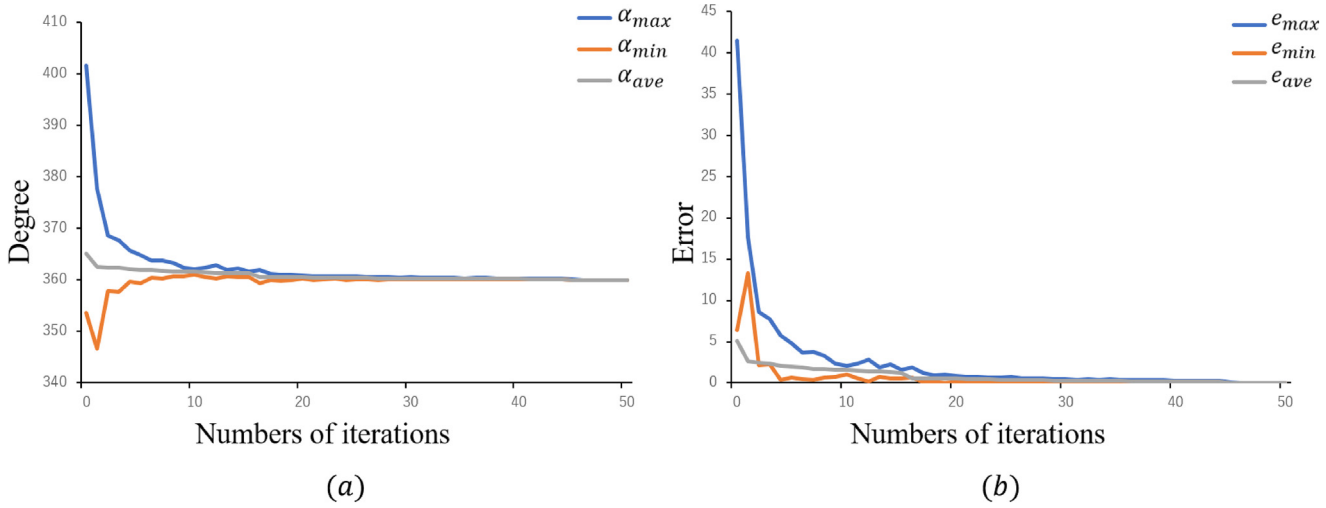

Fig. 6. Graphs of convergence created during optimization for producing developable surface.

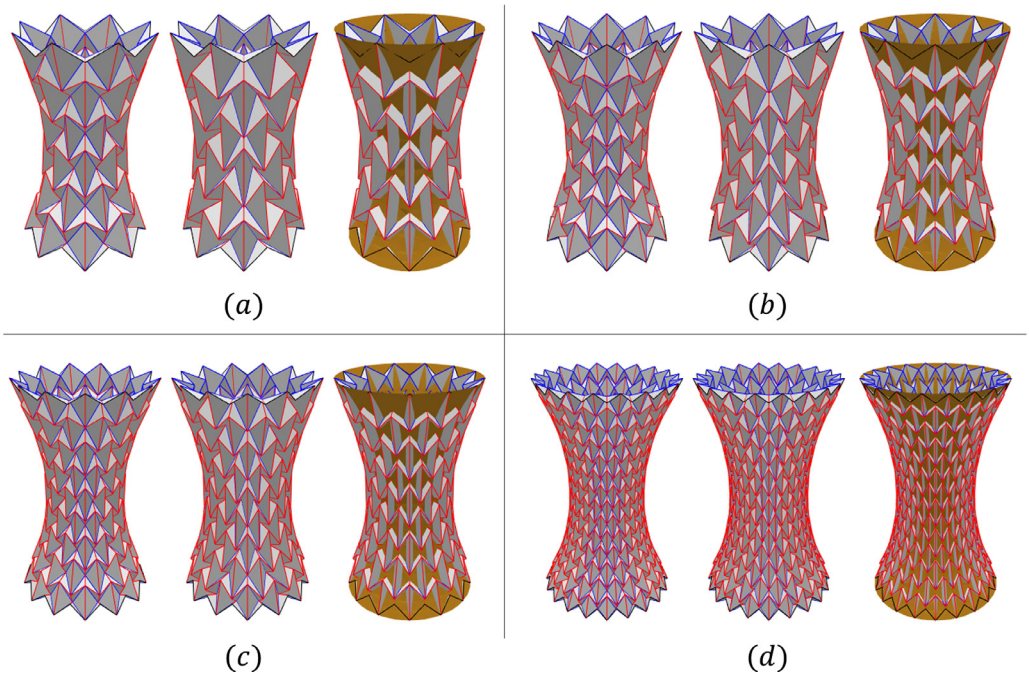

Fig. 7. Approximations with variable resolutions for same target surface.

Table 1
Parameters in detail and statistics of models shown in Fig. 7.

Approximations	N_u	N_v	Bases	e_{total}	$d(A,T)$	Time
(a)	8	6	48	9.68e−6	3.29e−2	0.31 min
(b)	10	7	70	6.54e−6	2.95e−2	0.85 min
(c)	13	9	117	9.80e−6	2.41e−2	9.35 min
(d)	20	14	280	9.94e−6	1.61e−2	158.60 min

where x and y denote vertices of the approximation A and the target surface T, respectively. $d(x,y)$ denotes the Euclid distance between x and y. $d(x,T)$ is the shortest distance between x and a set of y from T. $d(A,T)$ is similar in spirit to the Hausdorff distance, which is used to measure the difference between two surfaces. To compute $d(x,T)$, we sample all vertices from the approximation A for x by considering A is a discrete tessellation. When the target surface T is continuous, Dudte et al. (2016) use an optimization procedure to find the optimized u and v coordinates, which let the distance between y and x become shortest. Here, we densely sample a set of y by subdividing T, and then find the closest y for x. $d(A,T)$ is normalized by the diagonal length of the bounding box of the target surface T. Note that we are only concerned about the difference from A to T and do not measure the inverse distance $d(T,A)$; $d(A,T)$ and $d(T,A)$ are different because they are not symmetric. In Table 1, wenote that as the number of waterbomb bases increased, $d(A,T)$ decreased, which means that the result became closer to the target surface at the cost of more computational time.

We fabricated several approximations, shown in Fig. 8, where (a) shows a catenoid and (b) shows a cylinder. Both approximations contained 48 waterbomb bases. (c) shows a sphere containing 75 waterbomb bases, and (d) shows a vase containing 112 waterbomb bases. For each result shown in Fig. 8, we demonstrate a 3D model of the approximation, a crease pattern, and an origami piece.

We also approximated several 3D surfaces and show its crease pattern and rendered 3D model in Fig. 9, where (a) shows an approximation of a catenoid, (b) a sphere, (c) a cylinder, (d) a vase, (e) a torus, (f) a hyperbolic paraboloid, (g) a möbius strip. Details of the target surfaces are demonstrated in Table 2. Each surface ((a)–(e)) has an axisymmetric structure, and thus the boundary vertices along the left and right parts of the crease pattern are located at identical 3D positions to form the resulting approximation. Note that these vertices, which are used to connect the left and right parts of the crease pattern, have six adjacent facets in the 3D model. Therefore, we also applied our optimization process to these vertices in order to make them developable. For approximating torus, we not only connect the left and right parts of the crease pattern, but also the top and bottom parts (when N_v is even). As a result, we can generate a seamless approximation of torus (Fig. 9(e)). Additionally, we show an approximation of a hyperbolic paraboloid, which is the non-axisymmetric surface in Fig. 9(f), and an approximation of a Möbius strip, which is the non-orientable surface in Fig. 9(g). The Möbius strip approximation is not connected because the waterbomb bases at the start and end parts of the approximation had different orientations. Meanwhile, we demonstrate the detail results of the approximations (Fig. 9) in Table 3 correspondingly.

In terms of fabrication, folding a waterbomb tessellation is not an easy task because it requires multi-fold simultaneous actuation. The folding process becomes more difficult when the waterbomb tessellation contains more waterbomb bases. Pre-folding crease lines on a sheet of paper can alleviate this problem. However, the crease lines on paper become fuzzy after several pre-foldings. Therefore, we showed only crease patterns and rendered 3D models (Fig. 9) instead of results with folded paper. A more effective way for fabricating complex approximations with many waterbomb bases would be printing the crease patterns on a textile using polymers because a textile can be folded many times without obvious fatigue.

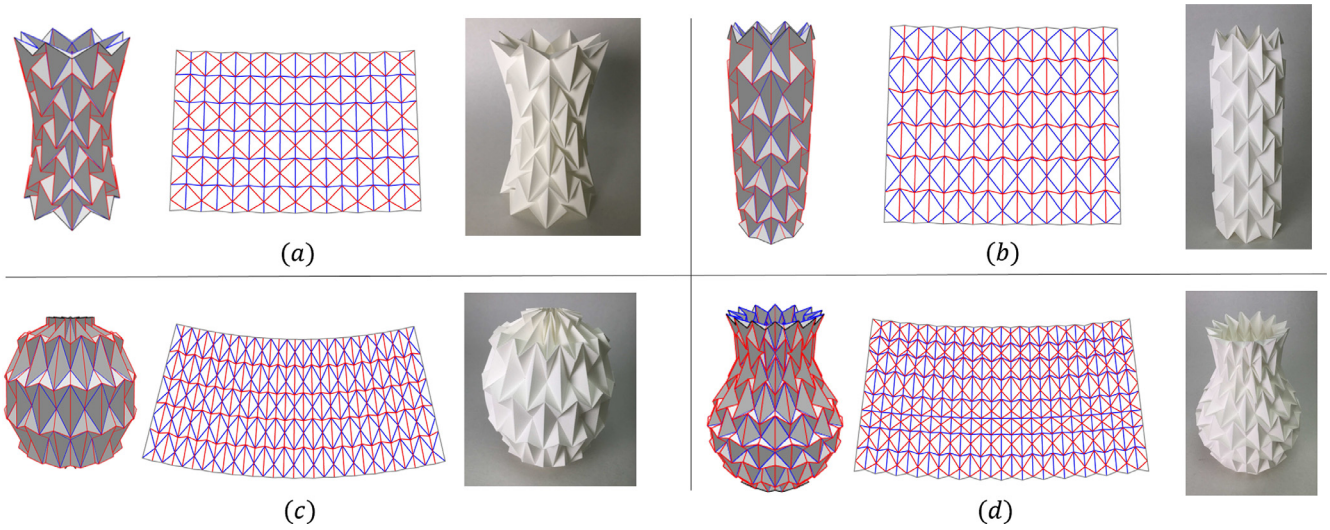

Fig. 8. Fabricated origami pieces.

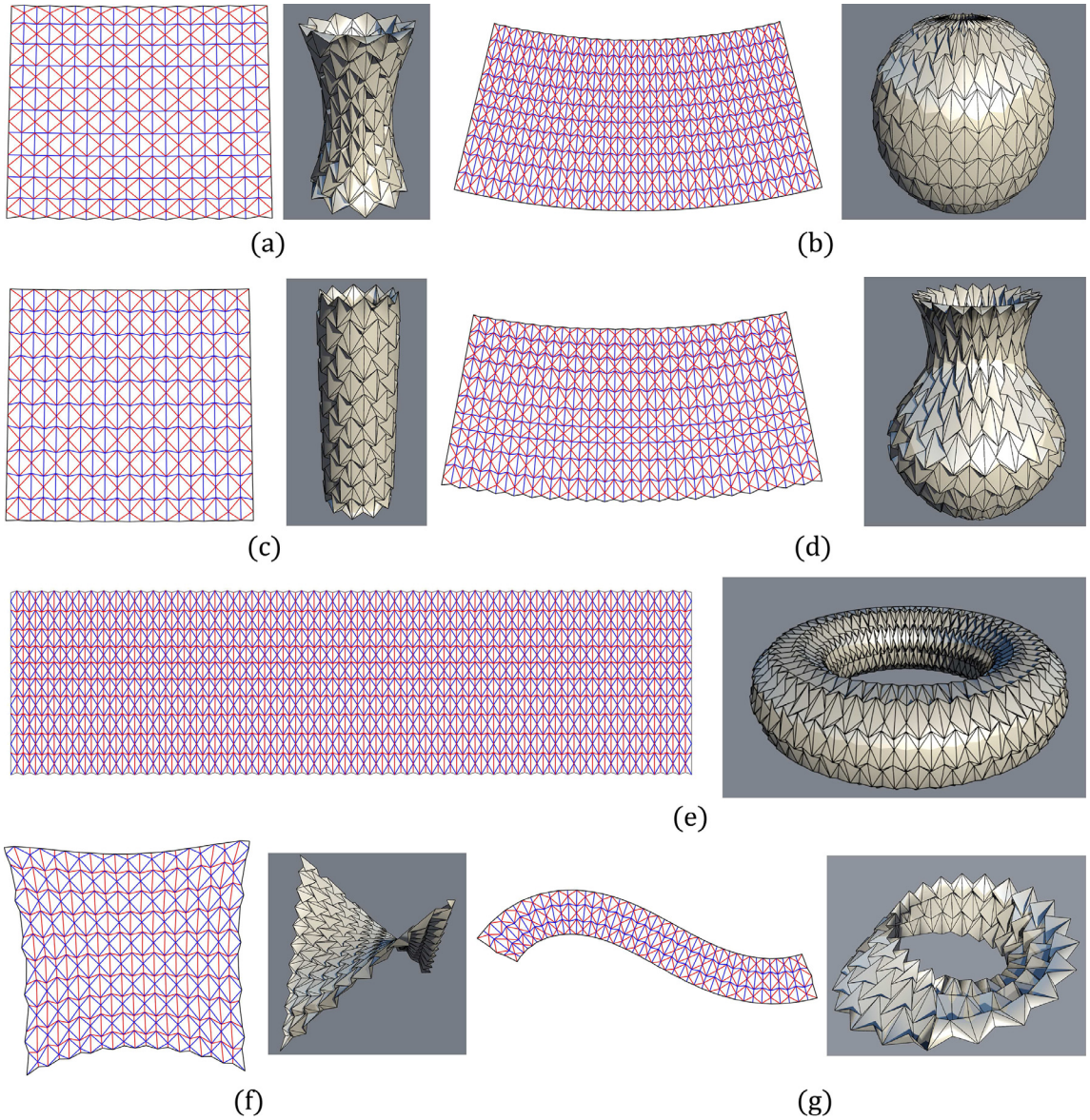

Fig. 9. Developable approximations consisting of generalized waterbomb tessellations.

Table 2
Target surfaces used for generating developable approximations shown in Fig. 9.

Targets	Equations (x, y, z)
(a)	$(\cosh\frac{v}{2.5}\cos u, \cosh\frac{v}{2.5}\sin u, v), u \in [0, 2\pi], v \in [-\pi, \pi]$
(b)	$(\cos v\cos u, \cos v\sin u, \sin v), u \in [0, 2\pi], v \in [\frac{-\pi}{2.2}, \frac{\pi}{2.2}]$
(c)	$(\cos u, \sin u, v), u \in [0, 2\pi], v \in [-\pi, \pi]$
(d)	$((2+\sin v)\cos u, (2+\sin v)\sin u, -v), u \in [0, 2\pi], v \in [-3, 4]$
(e)	$((3+\cos v)\sin u, (3+\cos v)\cos u, \sin v), u \in [0, 2\pi], v \in [-\pi, \pi]$
(f)	$(u, v, uv), u \in [-1, 1], v \in [-1, 1]$
(g)	$((1+\frac{v}{2}\cos\frac{u}{2})\cos u, (1+\frac{v}{2}\cos\frac{u}{2})\sin u, \frac{v}{2}\sin\frac{u}{2}), u \in [0, 2\pi], v \in [-1, 1]$

Table 3
Details of statistics of models shown in Fig. 9.

Approximations	N_u	N_v	Bases	e_{total}	$d(A,T)$	Time
(a)	10	10	100	$9.43e{-6}$	$2.60e{-2}$	4.79 min
(b)	25	10	250	$9.96e{-6}$	$1.44e{-2}$	152.27 min
(c)	10	10	100	$9.53e{-6}$	$1.59e{-2}$	2.58 min
(d)	21	10	210	$9.80e{-6}$	$1.59e{-2}$	74.09 min
(e)	55	10	550	$9.96e{-6}$	$8.97e{-3}$	853.18 min
(f)	10	10	100	$6.22e{-6}$	$1.35e{-2}$	2.35 min
(g)	22	3	66	$9.29e{-6}$	$1.40e{-2}$	0.44 min

5. Conclusion and future work

We proposed a method for approximating target surfaces, which are parametric surfaces of varying or constant curvatures, using generalized waterbomb tessellations. First, we described the generation of a base mesh by tiling the target surface using waterbomb bases. Then, we applied a simple numerical optimization algorithm to the base mesh to produce a developable approximation. Several developable approximations were presented to demonstrate the validity of our method. We provided a prototype system which enables us to interactively generate base meshes with variable resolutions and modify waterbomb bases.

Our work is different from *Origamizer* (Tachi, 2009, 2010a) and the system (Tachi, 2013), because ours is not based on the tucking technique, which hides unnecessary areas of a sheet of paper inside the shape. Our method is also differs from *Freeform Origami* (Tachi, 2010b), which generates a freeform surface by dragging the vertices of an origami in 3D. In addition, several existing approximating works were based on modified Miura-ori (Dudte et al., 2016; Song et al., 2017; Zhou et al., 2015), while we focus on the waterbomb tessellation, another basic origami tessellation, to fit on target surfaces. We have demonstrated that our method can tile waterbomb bases on target surfaces, which can be axisymmetric or non-axisymmetric as well as orientable or non-orientable.

As future work, three aspects of our study can be improved: (i) finding an optimal density to balance the approximation accuracy and amount of fabrication labor when generating a base mesh, (ii) achieving a developable approximation while restricting $d(A,T)$, that is, the distance between the resultant approximation A and target surface T, and (iii) generating flat-foldable and self-intersection-free approximations. Furthermore, we hope this work can be extended to approximate complex 3D models which can be parameterized into uv-plane and pave the way of fully solving the inverse-origami-design problem.

Conflict of interest

We wish to confirm that there are no known conflicts of interest associated with this publication.

References

Bateman, A. Paper mosaic origami tessellations. <http://www.papermosaics.co.uk/software.html>.

Chen, Y., Feng, H., Ma, J., Peng, R., & You, Z. (2016). Symmetric waterbomb origami. *Proceedings of the Royal Society A – Mathematical, Physical and Engineering Sciences, 472*, 20150846.

Dudte, L. H., Vouga, E., Tachi, T., & Mahadevan, L. (2016). Programming curvature using origami tessellations. *Nature Materials, 15*, 583–588.

Edwards, A., & Yan, H. (2014). Dna origami. In *Nucleic acid nanotechnology* (pp. 93–133). Springer.

Kuribayashi, K., Tsuchiya, K., You, Z., Tomus, D., Umemoto, M., Ito, T., et al. (2006). Self-deployable origami stent grafts as a biomedical application of ni-rich tini shape memory alloy foil. *Materials Science and Engineering: A, 419*(1), 131–137.

Lang, R. J. (1996). A computational algorithm for origami design. In *Proceedings of the twelfth annual symposium on computational geometry* (pp. 98–105). ACM.

Lang, R.J. Treemaker. <http://www.langorigami.com/article/treemaker/>.

Lang, R. J. (2009). Computational origami: From flapping birds to space telescopes. In *Proceedings of the twenty-fifth annual symposium on computational geometry* (pp. 159–162). ACM.

Lee, D.-Y., Kim, J.-S., Kim, S.-R., Koh, J.-S., & Cho, K.-J. (2013). The deformable wheel robot using magic-ball origami structure. In: *ASME 2013 international design engineering technical conferences and computers and information in engineering conference* (pp. V06BT07A040–V06BT07A040). American Society of Mechanical Engineers.

Meguro, T. (1991). The method to design origami. Origami Tanteidan Newspaper.

Mitani, J. (2009). A design method for 3D origami based on rotational sweep. *Computer-Aided Design and Applications, 6*(1), 69–79.

Mitani, J. (2011). A design method for axisymmetric curved origami with triangular prism protrusions. In *Origami 5: Fifth international meeting of origami science, mathematics, and education* (pp. 437–447). CRC Press.

Mitani, J. (2012). Column-shaped origami design based on mirror reflections. *Journal for Geometry and Graphics, 16*(2), 185–194.

Mitani, J., & Igarashi, T. (2011). Interactive design of planar curved folding by reflection. In *Pacific conference on computer graphics and applications-short papers, Kaohsiung, Taiwan, September, 2011* (pp. 21–23).

Miura, K. (1989). Map fold a la miura style its physical characteristics and application to the space science. *Research of Pattern Formation*, 77–90.

Nangreave, J., Han, D., Liu, Y., & Yan, H. (2010). Dna origami: A history and current perspective. *Current Opinion in Chemical Biology, 14*(5), 608–615.

Onal, C. D., Wood, R. J., & Rus, D. (2013). An origami-inspired approach to worm robots. *IEEE/ASME Transactions on Mechatronics, 18*(2), 430–438.

Pohl, D., & Wolpert, W. (2009). Engineered spacecraft deployables influenced by nature. *Advances in optomechanics* (Vol. 7424, pp. 742408). International Society for Optics and Photonics.

Rothemund, P. W. (2006). Folding dna to create nanoscale shapes and patterns. *Nature, 440*(7082), 297–302.

Song, K., Zhou, X., Zang, S., Wang, H., & You, Z. (2017). Design of rigid-foldable doubly curved origami tessellations based on trapezoidal crease patterns. *Proc. R. Soc. A* (Vol. 473, pp. 20170016). The Royal Society.

Tachi, T. (2009). 3D origami design based on tucking molecule. In R. Lang (Ed.), *The fourth international conference on origami in science, mathematics, and education, Pasadena* (pp. 259–272).

Tachi, T. (2010a). Origamizing polyhedral surfaces. *IEEE Transactions on Visualization and Computer Graphics, 16*(2), 298–311.

Tachi, T. (2010b). Freeform variations of origami. *Journal for Geometry and Graphics, 14*(2), 203–215.

Tachi, T., Masubuchi, M., & Iwamoto, M. (2012). Rigid origami structures with vacuumatics: Geometric considerations. In *Proc of the IASS-APCS Seoul, Korea, 21–24 May*.

Tachi, T. (2013). Freeform origami tessellations by generalizing reschs patterns. In *ASME 2013 international design engineering technical conferences and computers and information in engineering conference* (pp. V06BT07A025–V06BT07A025). American Society of Mechanical Engineers.

Tørring, T., Voigt, N. V., Nangreave, J., Yan, H., & Gothelf, K. V. (2011). Dna origami: a quantum leap for self-assembly of complex structures. *Chemical Society Reviews, 40*(12), 5636–5646.

Wilson, L., Pellegrino, S., & Danner, R. (2013). Origami sunshield concepts for space telescopes. AIAA Paper (2013-1594).

Zhao, Y., Kanamori, Y., & Mitani, J. (2018). Design and motion analysis of axisymmetric 3D origami with generic six-crease bases. *Computer Aided Geometric Design., 59*, 86–97.

Zhao, Y., Kanamori, Y., & Mitani, J. (2017). Geometry of axisymmetric 3D origami consisting of triangular facets. *Journal for Geometry and Graphics, 21*(1), 107–118.

Zhou, X., Wang, H., & You, Z. (2015). Design of three-dimensional origami structures based on a vertex approach. *Proc. R Soc. A* (Vol. 471, pp. 20150407). The Royal Society.