Pacific Graphics (2018)
H. Fu, A. Ghosh, and J. Kopf (Editors)

Short Paper

Mesh Parameterization : a Viewpoint from Constant Mean Curvature Surfaces

Hui Zhao [1] † Kehua Su [2] ‡ Chenchen Li [2] Boyu Zhang [5] § Shirao Liu [1] Lei Yang [1] Na Lei [4] Steven J. Gortler [5] ¶ Xianfeng Gu [3] ‖

[1] Capital Normal University, China [2] Wuhan University, China
[3] State University of New York at Stony Brook, USA [4] Dalian University of Technology, China [5] Harvard University, USA

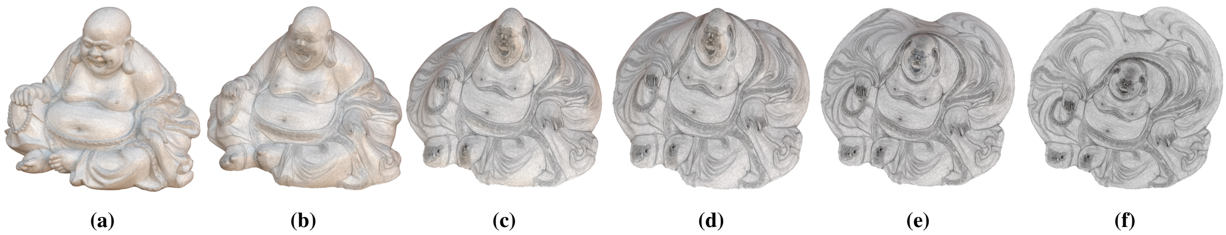


Figure 1: *The Buddha model deforms to a planar mesh.*

Abstract

We present a unified mesh paramterization algorithm for both planar and spheric domains based on mesh deformation. Unlike previous methods, our approach can produce intermediate frames from the original to target meshes. We derive and define a novel geometric flow: ***unit normal flow(UNF)*** *and prove that if unit normal flow converges, it will deform a surface to a constant mean curvature(CMC) surface, such as plane and sphere. Our method works by deforming meshes of disk topology to planes, meshes of spheric topology to spheres. The unit normal flow we propose also suggests a potential direction for creating CMC surfaces.*

CCS Concepts

•Computing methodologies → ***Shape modeling; Mesh models; Mesh geometry models;***

1. Introduction

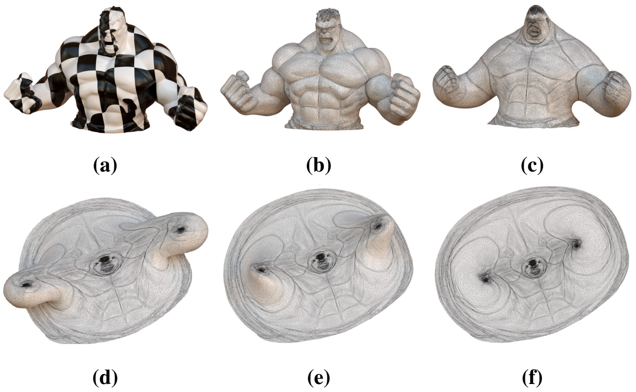
In this paper, we present a simple and novel algorithm of planar and spheric mesh parameterization. Our methodology is different from previous ones: we are not computing a direct embedding of a mesh onto planar or spheric domain, instead we deform it towards planar and spheric shape. Previous planar parameterization algorithms all have an implicit constraint: the target domain is the specific fixed $\{XY\}$ plane for all meshes. This limitation does not take the orientations and shapes of meshes into consideration. It also narrows the solution space and makes it difficult to achieve ideal solution. Our deformation based on approach unlocks this limitation, such that different shapes unfold naturally to the planes of varying orientations. Our method is more natural and intuitive.

Our method unifies planar and spheric mesh parameterization into a single framework, which consists of the iterations of two steps: average of face normals and surface deformation. For meshes of disk or sphere topology, they will converge to planar or spheric shapes automatically under the iterations. Figure 1, 2 and 3 demonstrate the deformations and their planar and spheric parameterizations respectively. Our approach produces bijective mappings in practice, although we do not prove it theoretically.

Firstly we compute the new normal of every point by averaging the normals of its neighbours. Secondly we reconstruct a surface which fits the current normals. The iterations of these two steps emerge a heat-like geometric flow on surfaces. We call it **unit normal flow(UNF)**. This observation guides the design of our algorithm. And our experiments on hundreds of discrete meshes suggest

† e-mail:alanzhaohui@qq.com
‡ e-mail:skh@whu.edu.cn
§ e-mail:bzhang@math.harvard.edu
¶ e-mail:sig@cs.harvard.edu
‖ e-mail:gu@cs.stonybrook.edu

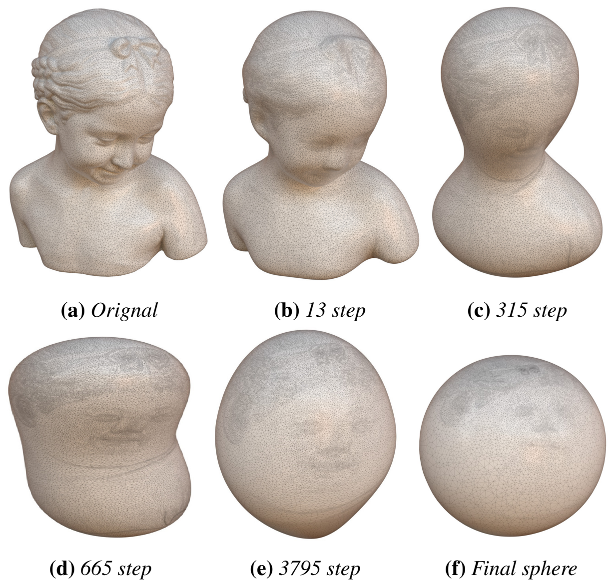
DOI: 10.2312/pg.20181272



(a) (b) (c)

(d) (e) (f)

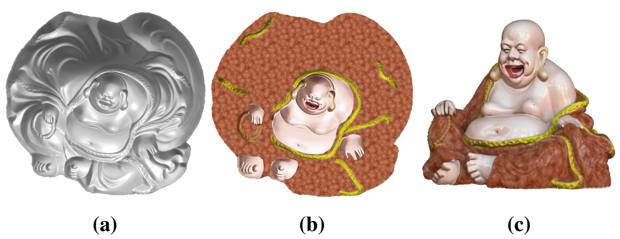
Figure 2: *(a) is the texturing; (b) is the original mesh; (c),(d),(e) are the intermediate deforming frames; (f) is the final planar mapping.*



(a) *Orignal* **(b)** *13 step* **(c)** *315 step*

(d) *665 step* **(e)** *3795 step* **(f)** *Final sphere*

Figure 3: *The deforamtion of bimba model and its spheric parameterization.*



(a) (b) (c)

Figure 4: *The texturing of the buddha model.*

that, as far as sphere and plane are concerned, the flow probably will converge.

The most important application of planar parameterization is texturing meshes. We render the planar mesh 1f with the normals in the corresponding original 3D mesh, such as shown in Figure 4a; then we draw textures on the rendered image 4b; finally the texturing 3D mesh is exhibited in Figure 4c. In our attached video, we also show the special effect of *two dimensional foil* described in the science-fiction books [Liu14, Liu15, Mor17].

In summary, our contributions in this paper are: 1) defining a novel geometric flow on surfaces: unit normal flow; 2) establishing and deriving the relationship between unit normal flow(UNF) and constant mean curvature(CMC) surfaces; 3) proposing a robust, simple-to-implement algorithm to discretizing and approximating the non-linear UNF; 4) applying the algorithm in the application of planar and spheric mesh parameterization, and our method has a special feature of mapping the selected partial parts of meshes onto a plane and keeping left parts unchanged.

2. Unit Normal Flow

Our motivation is deforming surfaces by the following criteria: The time derivatives of surface normals should be equal to the Laplacians of the normals. In this section, we define and propose **unit normal flow** mathematically. This flow is different from well-known mean curvature flow [KSBC12], averaged mean curvature flow [XPB06], Willmore flow [BS05, WBH*07], Ricci flow [JKLG08], surface diffusion flow [SK01, XPB06]. All these kinds of flows can be modelled as geometric partial differential equations(PDE) [XPB06, XZ08]. As far as we know, this definition is the first time to appear in the mathematical and graphic research literatures.

Let S be a smoothly immersed surface in $\mathbb{R}^3$. Let g be the metric on S restricted from $\mathbb{R}^3$. Let n be the smooth unit normal vector field on S. Denote $\langle , \rangle$ as the inner product and $\Delta_g n$ as the Laplacians of the unit normals. The formal definition of **unit normal flow** is the following:

$$\frac{dn}{dt} = \Delta_g n - \langle \Delta_g n, n \rangle \cdot n. \tag{1}$$

Notice that the norm of n is preserved under this flow, since $\frac{d}{dt}\langle n,n \rangle = \langle n, \frac{d}{dt}n \rangle = \langle n, \Delta_g n \rangle - \langle n, \Delta_g n \rangle = 0$

Lemma 1 If the Laplacians $\Delta_g n$ of the unit normal field n is parallel to n, i.e., $\Delta_g n // n$, then the mean curvature H of S is constant.

3. Our algorithms

The key point is the discreterization of the Laplace operator. In graphics community, the well-known cotangent Laplace operator [PP93] is for functions defined on vertice of meshes. Therefore it can not be used for our face normals. In this paper, we propose to use a simple method to approximate the Laplacian operator of normal functions on faces by the following formula:

$$\Delta_d n_i(t) = \sum_{j \in Neighbor(i)} n_j(t) - n_i(t) \approx \Delta_g n(t) \tag{2}$$

Where $n_j(t)$ denote the unit normal of face i at time t, $\Delta_d n_i$ represents the discrete Laplace operator of the face normal function, the

$Neighbor(i)$ denotes the neighbours of the face i, which includes i itself. Then the new normal at time $t+1$ is computed by the follows:

$$n_i(t+1) = \Delta_d n_i(t) + n_i(t) = \sum_{j \in Neighbor(i)} n_j(t) \quad (3)$$

Our flow is defined on **unit** face normals. Even though face normals n_i are unit, $\Delta_d n_i$ can not be guaranteed to be unit. We need normalize it. In practice, the faces in $Neighbor(i)$ are not constrained to be one-ring, they could be k-ring neighbours.

After the new face normals are computed in every step, we rotate all triangle faces from their old orientations to the current ones. However the triangles are rotated independently, the result triangle soup is not a valid mesh. We use the Poisson system based method [ZLL*17, ZG16] to reconstruct the triangles into a unified mesh. This step can be thought as solving a system of the unknown positions from the known normal variables .

In summary, our algorithm consists of two steps: in the first step, we average the unit face normals; in the second step, we deform or reconstruct the surface by the constraints of the current unit face normals. After the two steps, we get a new mesh which is smoother than previous one. These two steps are iteratively calculated until the flow converges and the shape of the mesh does not change anymore.

4. Experiments

To demonstrate the efficiency and robustness of our algorithm, we apply our algorithm to hundreds of challenging meshes of disk and spheric topology.

We demonstrate our mapping and corresponding texturing of disk-topology meshes in Figure 5. The intermediate deforming frames are also exhibited in Figure 1, 2.

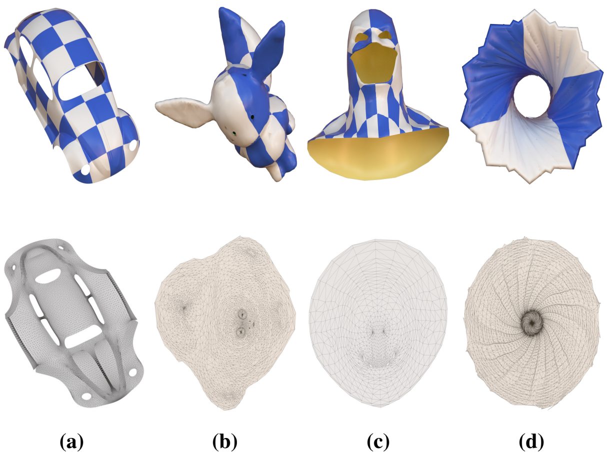


Figure 5: *The planar mapping and texturing of four multi-boundary meshes.*

The unfolded result of our algorithm is affected by the initial normals of boundary faces. When the boundary of a mesh is small and tight, possibly it can not unfolded towards a plane driven by its natural initial face normals. The mesh in Figure 6a will deform to the non-planar shape of Figure 6f after 500 hundreds of iterations through Figure 6d and 6e with its natural boundary face normals. We solve the problem by assigning it a set of specific boundary normals to pull faces apart, such that it is able to stretch to a plane in Figure 6c whose the corresponding texturing is shown in Figure 6b.

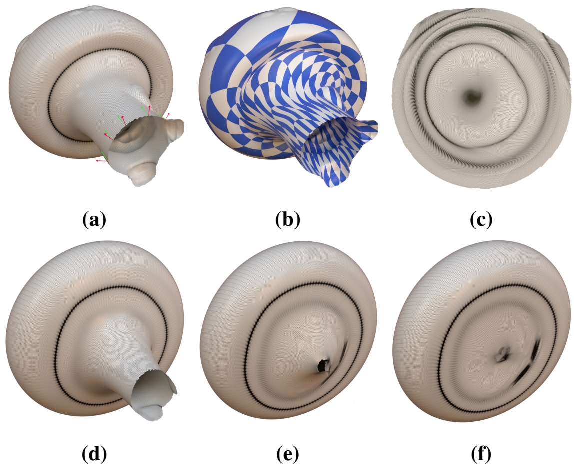


Figure 6: *The red arrows represent the natural face normals on the boundary of the original mesh (a) and the green ones are the normals we assigned; (d) (e) and(f) are the deforming meshes by red normals; (c) is the planar mesh deformed by the green normals; The texturing is exhibited in (b).*

In Figure 3, we demonstrates the deformation and parameterization of spheric topology. Unlike the planar one, the spheric unit normal flow converges slowly and needs thousands of iterations for most meshes.

Plane and sphere are special and simple constant mean curvature surfaces. Our approximation and discreterization of unit normal flow work successfully on them. For other kinds of CMC surfaces, our algorithm can also drive the flow to deform the corresponding discrete meshes. However, the convergent shapes are not CMC surfaces in exact mathematical sense. We call them CMC-like surfaces.

In Figure 7, we demonstrate the convergent shapes of the cylinders of a set of different radii and heights, constrained by two sets of the different boundary face normals, under our unit normal flow. In this experiment, the positions of the boundary vertice of the cylinders are fixed. The red arrows are the representatives of the first set of the boundary face normals; the green arrows are from the second set. The convergent shapes are catenoid-like surfaces, however the radii and heights we uses does not satisfy the exact mathematical formula of catenoids. In Figure 8, we deform a half-sphere and a unit disk to the different CMC-like surfaces under varying face normals constraints.

On one hand, CMC-like surfaces suggest and give us hint that unit normal flow could be mathematically convergent on smooth surfaces for all CMC surfaces. On the another hand, How to design

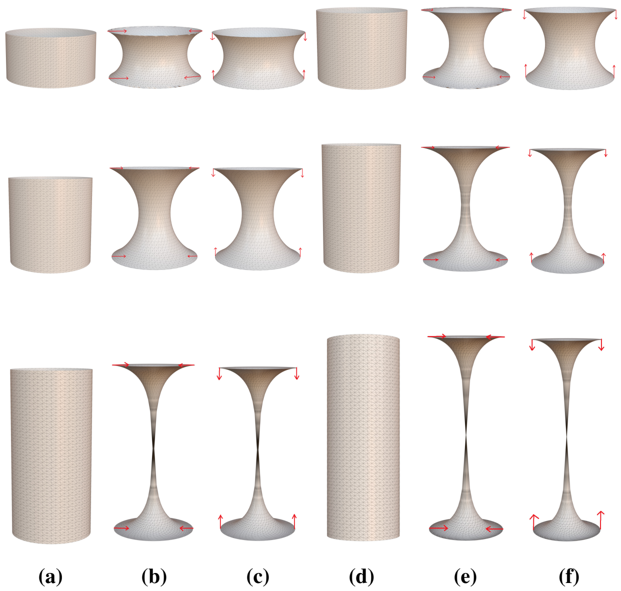


Figure 7: *The CMC-like surfaces generated by unit normal flow. The radii and heights of the cylinders are (10,10),(10,15),(10,20),(10,30),(10,40),(10,50) respectively.*

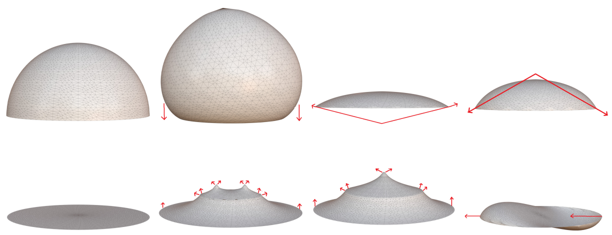


Figure 8: *The half-sphere and unit disk (a) are assigned three different set of boundary face normals; (b), (c), (d) show their corresponding convergent shapes.*

a more accurate discrete unit normal flow for other kinds of CMC surfaces is a challenging problem and our future works.

5. Conclusion and future work

We propose a special unit normal flow(UNF) to deform surfaces. This flow averages the normals of a smooth surface, and reconstruct the geometry to fit the smoothed normals. We define the mathematical equation of unit normal flow, and prove that the convergent surface has constant mean curvature if the flow is stable and converges. We also present an approximation method on discrete meshes and apply it to the applications of planar and spheric mesh parameterization. Our algorithm provides bijective mapping and it outperforms many state-of-art methods.

There are still some important works left for future. The convergence, singularity, existences and uniqueness of the unit normal flow are waiting to be proved. It is also a great challenge to design an efficient, stable and accurate discrete algorithm to construct other types of constant mean curvature surfaces besides planes and spheres.

Acknowledgements

We would like to thank anonymous reviewers for their insightful feedbacks, valuable comments, and suggestions. Some pictures are rendered by Mitsuba [Jak10]. All 3D models are from the AIM@SHAPE shape repository and Thingi10K repository. Thanks MeshDGP [Zha16] framework for the implementation reference. The project is partially supported by NSFC No. 61772379, 61772105, 61720106005, NSF DMS-1737812, NSF DMS-1418255, AFOSR FA9550-14-1-0193, NIH 1R01LM012434 and National ST Major Project of China (2018ZX10301201), the National Natural Science Foundation of China (11772047) and Key international collaborating Project from National Natural Science Foundation of China (11620101001).

References

[BS05] Bobenko A. I., Schröder P.: Discrete willmore flow. In *ACM SIGGRAPH 2005 Courses* (2005), ACM, p. 5.

[Jak10] Jakob W.: Mitsuba renderer, 2010. http://www.mitsuba-renderer.org.

[JKLG08] Jin M., Kim J., Luo F., Gu X.: Discrete surface ricci flow. *Visualization and Computer Graphics, IEEE Transactions on 14*, 5 (2008), 1030–1043.

[KSBC12] Kazhdan M., Solomon J., Ben-Chen M.: Can mean-curvature flow be modified to be non-singular? In *Computer Graphics Forum* (2012), vol. 31, Wiley Online Library, pp. 1745–1754.

[Liu14] Liu C.: *The three-body problem*, vol. 1. Macmillan, 2014.

[Liu15] Liu C.: *The dark forest*, vol. 2. Macmillan, 2015.

[Mor17] Morrison M. A.: Death's end. *World Literature Today 91*, 1 (2017), 74–75.

[PP93] Pinkall U., Polthier K.: Computing discrete minimal surfaces and their conjugates. *Experimental mathematics 2*, 1 (1993), 15–36.

[SK01] Schneider R., Kobbelt L.: Geometric fairing of irregular meshes for free-form surface design. *Computer aided geometric design 18*, 4 (2001), 359–379.

[WBH*07] Wardetzky M., Bergou M., Harmon D., Zorin D., Grinspun E.: Discrete quadratic curvature energies. *Computer Aided Geometric Design 24*, 8 (2007), 499–518.

[XPB06] Xu G., Pan Q., Bajaj C. L.: Discrete surface modelling using partial differential equations. *Computer Aided Geometric Design 23*, 2 (2006), 125–145.

[XZ08] Xu G., Zhang Q.: A general framework for surface modeling using geometric partial differential equations. *Computer Aided Geometric Design 25*, 3 (2008), 181–202.

[ZG16] Zhao H., Gortler S. J.: A report on shape deformation with a stretching and bending energy. *CoRR abs/1603.06821* (2016). URL: http://arxiv.org/abs/1603.06821.

[Zha16] Zhao H.: MeshDGP: A C Sharp mesh processing framework, 2016. http://meshdgp.github.io/.

[ZLL*17] Zhao H., Lei N., Li X., Zeng P., Xu K., Gu X.: Robust Edge-Preserved Surface Mesh Polycube Deformation. In *Pacific Graphics Short Papers* (2017), Barbic J., Lin W.-C., Sorkine-Hornung O., (Eds.), The Eurographics Association. doi:10.2312/pg.20171319.