A Shape-preserved Method to Improve the Developability of Mesh *

Zhixun Su*, Xiuping Liu, Xiaojie Zhou, Aihong Shen

Department of Applied Mathematics, Dalian University of Technology, Dalian 116024, China

Abstract

Developable surface plays an important role in computer aided design and manufacturing systems. This paper is concerned with improving the developability of mesh. Since subdivision is an efficient way to design complicated surface, we intend to improve the developability of the mesh obtained from Loop subdivision. The problem is formulated as a constrained optimization problem. The optimization is performed on the coordinates of the points of the mesh, together with the constraints of minimizing shape difference and maximizing developability, a developability improved mesh is obtained.

Keywords: Mesh manipulation; Developability; Constrained optimization

1 Introduction

Developability is one of the most important intrinsic properties of surface, which is applied widely in practice. For instance, the modeling of ship needs bending some plane material, while the material can not afford tensioning and bending infinitely, so the surface of the ship should be developable or approximatively developable. Another example is the design and manufacture of clothes and shoes. Thereby the study on developability of surface is necessary.

In the last few decades, much work has been down on the developability of two classes of surface: parameter surfaces and meshes. J. Lang ([1]) considered the rational Bezier surface and derive the condition formulation for rational Bezier surface. While the complexity of solving the condition formulation make it difficult to be applied to design developable Bezier surface in CAGD. T. Maekawa ([2]) proposed a kind of technique for designing developable B-spline surface of interpolating boundary curves. Other researches include methods based on projective geometry ([3]), optimization ([4]), and affine transformation ([5]). The properties of developable Bezier surface and the degree of freedom of developable surface design were also studied ([6]).

Due to the superiority in designing complex surface, triangle meshes as well as the developability of them have received much attention recently. L. P. Kobbelt ([8]) discussed the geometrical

*Project supported by the National Nature Science Foundation of China (No. 60275029).

*Corresponding author.

Email address: zxsu@comgi.com (Zhixun Su).

properties of meshes. C. L. Charlie ([7]) presented a developability-preserved FFD-based method. Other researchers focused on how to flatten a mesh into plane, energy-based ([10, 9]) and FEM-based methods ([11, 12]) were proposed to solve this problem.

The above mentioned mesh-based methods mainly discuss the approach to develop a mesh or to deform it with developability preserved, and seldom prior research on improving the developability of a mesh has been found in literature. Since subdivision is an important method for modeling complex objects, to improve the developability of subdivision surface becomes a valuable topic. This paper discuss how to improve the developability of subdivision surface. It is formulated as a constrained optimization problem, which can preserving the shape of the mesh simultaneously.

2 Mathematical Formulation

2.1 Developability of mesh

A surface is developable if it can be flattened into a plane without any distortion. At any regular point, the Gaussian curvature of a developable surface is identically zero. For a triangle mesh, the Gaussian curvature is zero expect on the vertices. So a developability of a triangle mesh is concerned with the vertices. The approximation Gaussian curvature function on a internal triangular node q_i is ([8])

$$k_{q_i} = (2\pi - \sum_k \theta_k)/(\frac{1}{3}\sum_k A_k)$$

where θ_k are the inner angles adjacent to q_i, and A_k are the corresponding triangle areas. The Gaussian curvature on a boundary vertex is zero. A internal vertex is called developable point when $\sum_j \theta_j = 2\pi$, otherwise, non-developable point. If all the internal vertices of a mesh M are developable points, the mesh is developable, otherwise, non-developable. So the developability of a mesh can be detected by the sum of the inner angular at every internal vertex [7]. While, simply stating whether a mesh is developability or not is insufficient to quantify the degree of developability of it. So we introduce the developability function of a mesh.

Definition [7] The developability function of a triangle mesh is defined as

$$D[M] = \frac{1}{A}\sum_i \delta(2\pi - \theta_{sum}(q_i))A_{q_i}$$

where $\delta(t)$ is the impulse function, $A_{q_i} = \frac{1}{3}\sum_j A_j$ is the sum area of the adjacent triangle of a vertex q_i on M with A_j being the area of the jth adjacent triangle corresponding to q_i, A is the area of the mesh M. $\theta_{sum}(q_i)$ is either the sum of inner angles adjacent to q_i when q_i is an internal vertex, or set to 2π when q_i is on the boundary of M.

$D[M]$ describe the degree of the developability of a mesh. When $D[M] = 1$, the mesh is developable. When $D[M] = 0$, all the internal vertices are non-developable points, which means the mesh has the lowest degree of developability.

2.2 Developability improving algorithm

For a given initial triangle mesh M, a subdivision surface M' can be obtained by Loop scheme. Suppose M' consists of n vertices, the coordinates of the n vertices form a n matrix

$$X = \begin{bmatrix} x_0 & x_1 & \cdots & x_{n-1} \\ y_0 & y_1 & \cdots & y_{n-1} \\ z_0 & z_1 & \cdots & z_{n-1} \end{bmatrix}^T$$

We intend to find a developable mesh $M^\star$ that means $D[M^\star] = 1$, and minimizing the difference between M and $M^\star$. The difference between M and $M^\star$ can be formulated as the following elastic energy function

$$E(X^\star) = \sum_i (\|q_{i,s}(X^\star)q_{i,t}(X^\star)\| - l_i^0)^2$$

where i is the index of a triangular edge, $q_{i,s} \in M'$ and $q_{i,t} \in M'$ are the vertices of the edge, and l_i^0 is the length of the triangular edge i on M'. $X^\star$ is the n matrix of the coordinates of the n vertices on $M^\star$

$$X^\star = \begin{bmatrix} x_0^\star & x_1^\star & \cdots & x_{n-1}^\star \\ y_0^\star & y_1^\star & \cdots & y_{n-1}^\star \\ z_0^\star & z_1^\star & \cdots & z_{n-1}^\star \end{bmatrix}^T$$

Therefore, we formulate the problem as a constrained optimization problem

$$\textbf{min } E(X^\star) \quad \textbf{s.t. } D[M^\star] = 1. \tag{1}$$

There is an impulse function in the definition of the developability function, which may lead to irregularity during the optimization. A new developability detect function G is defined to take place the developability function D

$$G[X] = \sum_i (g(q_i(X)))^2$$

where $q_i(X)$ is the position of a triangular vertex $q_i \in M$ determined by the parameter configuration X, and the function $g(q_i)$ is a vertex developability detect function

$$g(q_i) = \begin{cases} 0, & q_i \in B \\ 2\pi - \sum_k \theta_k, & \text{otherwise} \end{cases}$$

where B is the set of triangle vertices on the boundary of the given mesh M'.

We note that when $G[X^\star] = 0$, $D[M^\star] = 1$. Therefor, the constrained optimization problem Eq. (1) can be rewritten as

$$\textbf{min } E(X^\star) \quad \textbf{s.t. } G[X^\star] = 0. \tag{2}$$

3 Numerical Scheme

3.1 Conjugate gradient

The constrained optimization Eq. (2) can be converted into a unconstrained optimization problem

$$\Phi(X^\star) = E(X^\star) + \frac{\rho}{2}(G(X^\star))^2 \tag{3}$$

where $\frac{\rho}{2}(G(X^\star))^2$ is the penalty term. The choice of the ρ is not trivial, in our algorithm,

$$\rho = \frac{1}{m_e(G[X^{(k)}])^2}\sum_i (l_i^0)^2$$

where m_e is the number of triangular edges for the kth iteration.

We apply the Conjugate gradient to solve Eq. 2, the algorithm is

$$\left.\begin{array}{l} X^{(k+1)} = X^{(k)} - \alpha_k p^{(k)}, \\ \Phi(X^{(k+1)}) = \min {}_\alpha \Phi(X^{(k)} - \alpha p^{(k)}), \\ p^{(k+1)} = \nabla\Phi(X^{(k+1)}) - \beta_k p^{(k)}, \\ \beta_k = \dfrac{\| \nabla \Phi(X^{(k+1)})\|^2}{\| \nabla \Phi(X^{(k)})\|^2}, \\ p^{(0)} = \nabla\Phi(X^{(0)}). \end{array}\right\}$$

3.2 Computation of the gradient

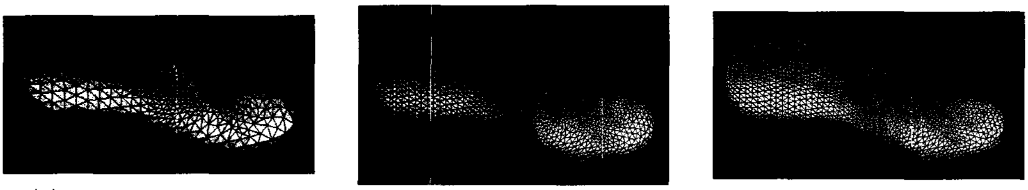
In the above algorithm, an important process is the computation of the gradient of the objective function. We will give the gradients of E and G respectively. Since E is concerned with the vertices q_v as $E(q_v) = \sum(\|q_v q_i\| - l_{vi})^2$, where q_i are the adjacent vertices of q_v. Thereby gradient of E with respect to q_v is

$$\frac{\partial E}{\partial q_v} = \frac{E(q_v + h) - E(q_v - h)}{2h}.$$

And similarly, the gradient of G with respect to q_v is

$$\frac{\partial G}{\partial q_v} = \frac{(g(q_v + h))^2 - (g(q_v - h))^2}{2h}$$

where h is the time step.

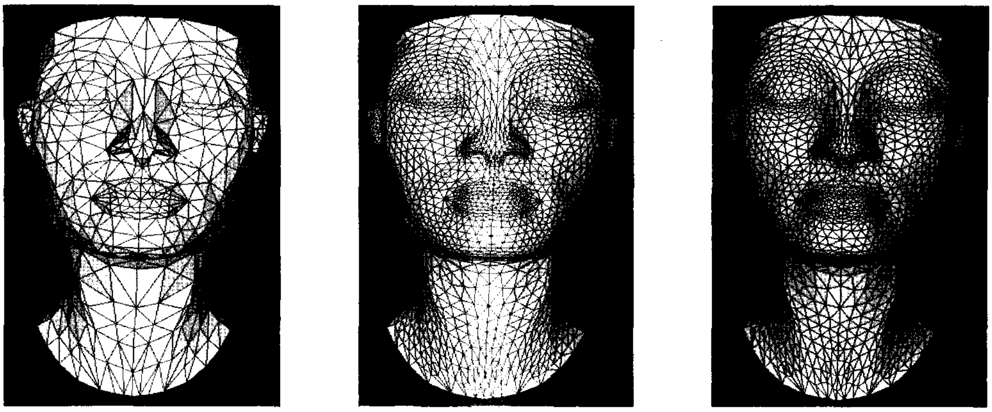


(a) Initial control mesh (b) Loop subdivision surface (c) Mesh after optimization

Fig. 1: The optimization of the shoe mesh

4 Experimental Result

In this section we give some examples of the present method (Fig. 1, 2), and table 1 lists the statistics of them. In table 1, $D[M_0]$, $D[M_L]$ and $D[M_R]$ denote the developability of the initial control mesh, the mesh after Loop subdivision, and the mesh after optimization. δS denotes the relative difference between the Loop subdivision surface and the mesh after optimization $\sum_i |l_i - l_i^0| / \sum_i l_i^0$, where l_i^0 and l_i are the lengths of the edge on the Loop subdivision surface and the corresponding edge on the mesh after optimization. δD denotes the relative improvement of the developability between the Loop subdivision surface and the mesh after optimization.



(a) Initial control mesh (b) Loop subdivision surface (c) Mesh after optimization

Fig. 2: The optimization of the face

Table 1: The statistics of the computation

Example	$G[M_0]$	$G[M_L]$	$G[M_R]$	δS	δD
Fig. 1	0.459899	0.687809	0.693811	0.1407%	0.8726%
Fig. 2	0.153149	0.307269	0.309201	1.0631%	0.6288%

5 Conclusion

In this paper, a shape-preserved developability improvement method for mesh is presented based on optimization. Numerical experiments illustrate its feasibility.

We find that for a fine mesh, the developability will be improved through Loop subdivision, while this does not necessary hold for a relative coarse mesh. Therefor, the relationship between subdivision and the developability of a mesh should be studied in the future.

References

[1] J. Lang and O. Röschel, Developable (1,n)-Bezier surface, Computer Aided Geometric Design 9 (1992) 291-298.

[2] T. Maekawa and B. Chalfant, Design and tesselation of B-spline developable surface, ASME Transaction Journal of Mechanical Design 120 (1998) 453-61.

[3] H. Pottmann and G. Farin, Developable rational Bezier and B-spline surfaces, Computer Aided Geometric Design 12 (1995) 513-531.

[4] F. C. Park, Jungyun Yu and Changmook Chun, Design of developable surfaces using optimal control, Trans. ASME J. Mech. Design 124 (2002) 602-608.

[5] Günter Aumann, A simple algorithm for designing developable Bezier surfaces, Computer Aided Geometric Design 20 (2003) 601-619.

[6] Chih-Hsing Chu and Carlo H. Séquin, Developable Bezier patches: properties and design, Computer Aided Design 34 (2002) 511-527.

[7] Charlie C. L. Wang et al., Developability-preserved free-form deformation of assembled patchs, ACM Symposium on Solid modeling and Application, Genova, Italy, pp. 231-236, 2004.

[8] L. P. Kobbelt et al., Geometric modeling based on polyonal meshes, EUROGRAPHICS 2000 Tutorial.

[9] Charlie C. L. Wang, Shana S-F. Smith and Matthew M. F. Yuen, Surface flattening based on energy model, Computer Aided design 34 (2002) 823-833.

[10] J. McCartney, B. K. Hinds and B. L. Seow, The flattening of triangulated surfaces incorporating darts and gussets, Computer Aided Design 31 (1999) 249-260.

[11] T. Shimada and Y. Tada, Approximate transformation of an arbitrary curved surface into a plane using dynamic programming, Computer Aided Design 23 (2) (1991) 153-159.

[12] Qi-Lin Zhang and Xiao-Qun Luo, Finite element method for developing arbitrary surfaces to flattened forms, Finite Elements in Analysis and Design 39 (2003) 977-984.