Interpolating Meshes of Arbitrary Topology

Graphics & Geometric Modeling Lab

Abstract. A new method for constructing a smooth surface that interpolates the vertices of an arbitrary mesh is presented. The mesh can be open or closed. Normals specified at vertices of the mesh can also be interpolated. The interpolating surface is obtained by locally adjusting the limit surface of the given mesh (viewed as the control mesh of a Catmull-Clark subdivision surface) so that the modified surface would interpolate all the vertices of the given mesh. The local adjustment process is achieved through locally blending the limit surface with a surface defined by non-uniform transformations of the limit surface. This local blending process can also be used to smooth out the shape of the interpolating surface. Hence, a *surface fairing* process is not needed in the new method. Because the interpolation process does not require solving a system of linear equations, the method can handle meshes with large number of vertices. Test results show that the new method leads to good interpolation results even for complicated data sets. The new method is demonstrated with the Catmull-Clark subdivision scheme. But with some minor modification, one should be albe to apply this method to other subdivision schemes as well.

CR Categories: I.3.5 [Computer Graphics]: Computational Geometry and Object Modeling - curve, surface, solid and object representations;

Keywords: subdivision, subdivision surfaces, Catmull-Clark subdivision surfaces, interpolation

1 Introduction

Constructing a smooth surface to interpolate the vertices of a given mesh is an important task in many areas, including geometric modeling, computer graphics, computer animation, interactive design, and scientific visualization. The interpolating surface sometime is also required to interpolate normal vectors specified for some or all of the mesh vertices. Developing a general solution for this task is difficult because the required interpolating surface could be of arbitrary topology and with arbitrary genus. Traditional representation schemes such as B-spline or Bézier surfaces can not represent such a complex shape with only one surface.

Subdivision surfaces were introduced as an efficient technique to model complex shapes [2][3][10]. But building a connection between a given mesh and an interpolating subdivision surface has never really been successful when the number of vertices of the given mesh is large [1]. One exception is a work published recently [11]. In this paper, an iterative interpolation technique similar to the one used in [8] for non-uniform B-spline surfaces is proposed for subdivision surfaces. Since the iterative approach does not require solving a system of linear equations, it can handle meshes with large number of vertices. But the paper fails to prove the convergence of the iterative process.

In this paper we will address the problem of 'constructing a smooth surface to interpolate the vertices of a given mesh' and present a new solution to this problem. We briefly review previous work in this area first.

1.1 Previous Work: A Brief Review

There are two major ways to interpolate a given mesh with a subdivision surface: *interpolating subdivision* [4, 6, 7, 15, 20] or *global optimization* [5, 13]. In the first case, a subdivision scheme that interpolates the control vertices, such as the Butterfly scheme [4], Zorin et al's improved version [20] or Kobbelt's scheme [7], is used to generate the interpolating surface. New vertices are defined as local affine combinations of nearby vertices. This approach is simple and easy to implement. It can handle meshes with large number of vertices. However, since no vertex is ever moved once it is computed, any distortion in the early stage of the subdivision will persist. This makes interpolating subdivision very sensitive to irregularity in the given mesh. In addition, it is difficult for this approach to interpolate normals or derivatives.

The second approach, *global optimization*, usually needs to build a global linear system with some constraints [14]. The solution to the global linear system is a control mesh whose limit surface interpolates

[1]Interpolating subdivision [4] will be addressed shortly

the vertices of the given mesh. This approach usually requires some fairness constraints in the interpolation process, such as the energy functions presented in [5], to avoid undesired undulations. Although this approach seems more complicated, it results in a traditional subdivision surface. For example, the method in [5] results in a Catmull-Clark subdivision surface (CCSS), which is C^2-continuous almost everywhere and whose properties are well studied and understood. The problem with this approach is that a global linear system needs to be built and solved. It is difficult for this approach to handle meshes with large number of vertices.

There are also techniques that produce surfaces to interpolate given curves or surfaces that near- (or quasi-) interpolate given meshes [9]. But those techniques are either of different natures or of different concerns and, hence, will not be discussed here.

1.2 Overview

In this paper a new method for constructing a smooth surface that interpolates the vertices of a given mesh is presented. The mesh can be of arbitrary topology and can be open or closed. Normal vectors specified for any vertices of the mesh can also be interpolated. The basic idea is to view the given mesh as the control mesh of a Catmull-Clark subdivision surface and locally adjust the limit surface of the given mesh so that the resulting surface would not only interpolate vertices of the given mesh, but also possess a satisfactory smooth shape. The local adjustment process is achieved through blending the limit surface S with a blending surface T defined by non-uniform transformations of the limit surface. By performing the blending process at different selected points, we are able to (1) ensure the modified surface would interpolate the given mesh, (2) prevent it from generating unnecessary undulations, and (3) smooth out the shape of the resulting surface.

The new method has two main advantages. First, since we do not have to compute the interpolating surface's control mesh, there is no need to solve a system of linear equations. Therefore, the new method can handle meshes with large number of vertices, and is more robust and stable. Second, because the local blending process can be used to smooth out the shape of the interpolating surface, a *surface fairing* process is not needed in the new method.

An example of this interpolation process is shown in Figure **??**. The surfaces shown in Figures **??**, **??** and **??** all interpolate the mesh shown in Figure **??**. The blending areas in Figure **??** are automatically selected by the system while Figures **??** and **??** have user selected blending areas in the upper portion and lower portion of the teapot body afterward. It is easy to see from Figure **??** that local control is necessary when better quality interpolating surfaces are needed.

The new method is demonstrated with Catmull-Clark subdivision surfaces here (by viewing the given mesh as the control mesh of a Catmull-Clark subdivision surface). But with a minor modification, one should be able to apply it to other subdivision schemes as well.

The remaining part of the paper is arranged as follows. In Section 2, the basic idea of our locally controllable interpolation technique for closed meshes is presented. The construction process of a blending surface is presented in Section 3. In Section 4, a local parametrization is introduced. The blending process around an extraordinary point or an arbitrarily selected point is discussed in Section 5 and Section 6, respectively. Issues on dealing with normal interpolation and handling open meshes are discussed in Section 7 and Section 8, respectively. Implementation issues and test results are presented in Section 9. Concluding marks are given in Section 10.

2 Basic Idea

Given a mesh M and a subdivision scheme, our task is to find a smooth subdivision surface to interpolate M. We use the following notations in the paper: A refers to the matrix that calculates all the limit points of M with respect to the given subdivision scheme, $I(M)$ refers to the subdivision surface that interpolates M, $S(M)$ refers to the limit surface of M, and $L(M)$ refers to the limit points of M. Note that $I(M)$ and $S(M)$ are surfaces and $L(M) = A * M$ is a mesh of the same topology as M. Without loss of generality, we shall assume the subdivision scheme considered here is the Catmull-Clark scheme. But the concept works for all subdivision schemes.

Let M_0 be the given mesh. Then the task is to find $I(M_0)$, a Catmull-Clark subdivision surface that interpolates the vertices of M_0. If we can find an *offset surface* R that moves $S(M_0)$, the Catmull-Clark subdivision surface of M_0, to $I(M_0)$ everywhere, i.e.,

$$R + S(M_0) = I(M_0)$$

then the interpolation problem is solved. The question is, how should R be constructed?

$S(M_0)$ can be considered as a Catmull-Clark surface that interpolates $L(M_0)$, i.e., $S(M_0) = I(L(M_0)) = I(A * M_0)$. To move $S(M_0)$ to $I(M_0)$ everywhere, T_1 must be able to make up the difference between $L(M_0)$

and M_0. A natural choice is to define T_1 as an interpolating Catmull-Clark subdivision surface of M_1, the difference between M_0 and $L(M_0)$. Hence, by replacing T_1 with $I(M_1)$ in the above equation, we have the following recurrence formula

$$I(M_1) + S(M_0) = I(M_0)$$

where

$$M_1 = M_0 - L(M_0)$$

M_1 has the same topology as M_0, hence $I(M_0)$ and $I(M_1)$ can be constructed exactly the same way. By repeating the recurrence formula for $i = 1, 2, \dots,$ we get a sequence of meshes M_i $(1 \le i \le \infty)$ such that

$$I(M_{i+1}) + S(M_i) = I(M_i)$$

and

$$M_{i+1} = M_i - L(M_i). \quad (1)$$

Consequently, we have

$$I(M_0) = \sum_{i=0}^{n} S(M_i) + I(M_{n+1})$$

and

$$M_i = (E - A)^i M_0$$

where E is the identity matrix and A is the matrix that calculats all the limit points of the given matrix M_0. It can be shown that $(E - A)^i$ converges to zero when i tends to infinity (see Appendix for a proof). Hence, we have

$$\lim_{n\to\infty} I(M_{n+1}) = \mathbf{0}$$

On the other hand, because A is invertible (see Appendix for a proof), we have

$$\sum_{i=0}^{n} S(M_i) = S(\sum_{i=0}^{n} M_i) = S(A^{-1}(E-(E-A)^{n+1})M_0)$$

By combining the above two equations, we have

$$I(M_0) = S(\sum_{i=0}^{\infty} M_i) = S(A^{-1}M_0) \quad (2)$$

If we define

$$\hat{M} = \sum_{i=0}^{\infty} M_i \quad (3)$$

then $\hat{M} = A^{-1}M_0$ holds as well. Hence $I(M_0)$ is also a subdivision surface and $\hat{M}$ is the control mesh of $I(M_0)$. Traditionally, people find $\hat{M}$ by solving the system of linear equations $A\hat{M} = M_0$ directly [5, 13]. It is difficult to use this approach to deal with meshes with large number of vertices. With eqs. (2) and (3), this is not a problem any more because $\hat{M}$ can be obtained by iteratively applying eq. (1) to get enough terms in (3) until a desired precision is reached. This approach effectively reduces a global problem to a local problem because eq. (1) is performed on the basis of individual vertices. More importantly, just like Fourier transformation, a subdivision surface can be represented as the sum of an infinite series of subdivision surfaces. For example, for a given control mesh M, $S(M)$ can be represented as

$$S(M) = I(L(M)) = S(\sum_{i=0}^{\infty} M_i)$$

where $M_0 = L(M)$ and M_i are defined by eq. (1) for $i \ge 1$. This property can be used in applications such as fairing, smoothing, sharpening, low pass or high pass filtering, etc.

3 Summary

A new interpolation method for meshes with arbitrary topology is presented. The interpolation process is a local process, it does not require solving a system of linear equations. Hence, the method can handle data set of any size.

The interpolating surface is obtained by locally adjusting the limit surface of the given mesh (viewed as the control mesh of a Catmull-Clark subdivision surface) so that the modified surface interpolates all the vertices of the given mesh. This local adjustment process can also be used to smooth out the shape of the interpolating surface. Hence, a *surface fairing* process is not needed in the new method.

The new method can handle both open and closed meshes. It can interpolate not only vertices, but normals and derivatives as well. These normals and derivative can be anywhere, not just at the vertices of the given mesh. Test results show that the new method leads to good interpolation results even for complicated data sets.

The resulting interpolating surface is not a Catmull-Clark subdivision surface. It does not even satisfy the convex hull property [19]. But the resulting interpolating surface is guaranteed to be C^2 continuous everywhere except at some extraordinary points, where it is C^1 continuous. Using a technique similar to the one presented in [19], a C^2 continuous interpolating surface can also be achieved.

References

[1] Barsky B A, End conditions and boundary conditions for uniform B-spline curve and surface representation, *Computers in Industry*, 1982, 3(1/2):17-29.

[2] Catmull E, Clark J, Recursively generated B-spline surfaces on arbitrary topological meshes, *Computer-Aided Design*, 1978, 10(6):350-355.

[3] Doo D, Sabin M, Behavior of recursive division surfaces near extraordinary points, Computer-Aided Design, 1978, 10(6):356-360.

[4] Dyn N, Levin D, and Gregory J A, A butterfly subdivision scheme for surface interpolation with tension control, *ACM Transactions on Graphics*, 1990, 9(2):160-169.

[5] Halstead M, Kass M, DeRose T, Efficient, fair interpolation using Catmull-Clark surfaces, *ACM SIGGRAPH*, 1993:35-44.

[6] Levin A, Interpolating nets of curves by smooth subdivision surfaces, *ACM SIGGRAPH*, 1999, 57-64.

[7] Kobbelt L, Interpolatory subdivision on open quadrilateral nets with arbitrary topology, *Computer Graphics Forum*, Eurographics, V.15, 1996.

[8] Lin H, Wang G, Dong C, Constructing Iterative Non-Uniform B-spline Curve and Surface to Fit Data Points, *SCIENCE IN CHINA*, Series F, 2004, 47:315-331.

[9] Litke N, Levin A, Schröder P, Fitting subdivision surfaces, *Proceedings of the conference on Visualization* 2001:319-324.

[10] Loop C T, Smooth subdivision surface based on triangle, *Master's thesis*, Department of Mathematics, University of Utah, 1987.

[11] Maekawa T, Matsumoto Y, Namiki K, Interpolation by geometric algorithm, *Computer-Aided Design*, 2007, 39:313-323.

[12] Magnus IR, Neudecker H, *Matrix Differential Calculus with Applications in Statistics and Econometrics*, John Wiley and Sons, 1988, New York.

[13] Nasri A H, Surface interpolation on irregular networks with normal conditions, *Computer Aided Geometric Design*, 1991, 8:89-96.

[14] Nasri A H, Sabin M A, Taxonomy of interpolation constraints on recursive subdivision curves, *The Visual Computer*, 2002, 18(4):259-272.

[15] Schaefer S, Warren J, A Factored Interpolatory Subdivision Scheme for Quadrilateral Surfaces, *Curves and Surface Fitting*, 2002, 373-382.

[16] Stam J, Exact Evaluation of Catmull-Clark Subdivision Surfaces at Arbitrary Parameter Values, *Proceedings of SIGGRAPH* 1998:395-404.

[17] Stam J, Evaluation of Loop Subdivision Surfaces, *SIGGRAPH'99 Course Notes*, 1999.

[18] Reif, U. A unified approach to subdivision algorithms near extraordinary points. *Computer Aided Geometric Design* 12, 2, 153C174, 1995.

[19] Adi Levin, Modified Subdivision Surfaces with Continuous Curvature, *Proceedings of SIGGRAPH*, 1035-1040, 2006.

[20] D. Zorin, P. Schröder, W. Sweldens, Interpolating Subdivision for meshes with arbitrary topology, *ACM SIGGRAPH*, 1996:189-192.

[21] Kestutis Karciauskas and Jörg Peters, Guided Subdivision, http://www.cise.ufl.edu/research/SurfLab/papers/05guiSub.pdf, 2005.

[22] Warren, J., and Weimer, H., *Subdivision Methods for Geometric Design*, Morgan Kaufmann, 2002.

Appendix

A. Proof of convergence of $(E - A)^i$

To prove this, we just need to show the eigen values λ_i of A are all positive but smaller than or equal to 1. Here we present the proof for Catmull-Clark subdivision scheme only. Other schemes can be proved similarly. The assumption is the given mesh M has at least one vertex with a valence bigger than 3. Note that with all the eigen values being positive, the matrix A is invertible.

First, we perfrom a subdivision on the given mesh M to ensure each face is 4-sided. We use V, E_i and F_i to denote vertex points, edge points and face points of the given mesh and v, e_i and f_i to denote vertex points, edge points and face points in the new mesh. They satisfy the following formulas:

$$\begin{aligned} v &= \frac{n-2}{n}V + \frac{1}{n^2}\sum_{i=1}^{n} E_i + \frac{1}{n^2}\sum_{i=1}^{n} f_i \\ e_i &= \frac{V + E_i + f_i + f_{i+1}}{4} \\ f_i &= \frac{V + E_i + E_{i+1} + F_1^i + \cdots + F_{m_i-3}^i}{m_i} \end{aligned} \quad (4)$$

where n is the valence of V and m_i is the number of vertices of the face that contains V, E_i and E_{i+1}.

Since each face after the subdivision is 4-sided, we can use the following formula [5] to calculate the limit point of v on the limit surface $S(M)$:

$$v^\infty = \frac{n^2}{n(n+5)}v + \frac{4}{n(n+5)}\sum_{i=1}^{n} e_i + \frac{1}{n(n+5)}\sum_{i=1}^{n} f_i$$

The relationship between this limit point and vertices of the original mesh M can be obtained by replacing the new vertex point, edges points and face points with the corresponding items in eq.(4). We have

$$\begin{aligned} v^\infty = \frac{4}{n(n+5)}((\frac{n(n-1)}{4} + (\sum_{i=1}^{n}\frac{1}{m_i}))V \\ + \sum_{i=1}^{n}(\frac{1}{2} + \frac{1}{m_i} + \frac{1}{m_{i-1}})E_i \\ + \sum_{i=1}^{n}\frac{F_1^i + \cdots + F_{m_i-3}^i}{m_i}) \end{aligned} \quad (5)$$

The matrix A is defined by eq. (5). A is of dimension $K \times K$ where K is the number of vertices of M. A can be written as

$$A = DS$$

where D is a diagonal matrix of the following form

$$D = \begin{bmatrix} \frac{4}{n_1(n_1+5)} & 0 & \cdots & 0 \\ 0 & \frac{4}{n_2(n_2+5)} & \cdots & 0 \\ \vdots & \vdots & & \vdots \\ 0 & 0 & \cdots & \frac{4}{n_K(n_K+5)} \end{bmatrix}$$

and S is a symmetric, simi-positive definite matrix. n_i in D is the valence of vertiex V_i in the given matrix M.

To prove the symmetry of S, note that the value of a non-zero, off-diagonal entry of S is either of the form $(\frac{1}{2} + \frac{1}{m_k} + \frac{1}{m_{k-1}})$ or $\frac{1}{m_k}$. If the value of a non-zero, off-diagonal entry a_{ij} is of the first form, it means there is an edge between the vertices V_i and V_j in the original mesh and the numbers of vertices in the two faces that share this edge are m_k and m_{k-1}, respectively. This holds either $i > j$ or $j > i$. Hence we must have $a_{ij} = a_{ji}$. If the value of a non-zero, off-diagonal entry a_{ij} is of the second form, it means there is a face in the original mesh that contains V_i and V_j as non-adjacent vertices and the number of vertices of the face is m_k. Again, this hold either $i > j$ or $j > i$. Hence, we must have $a_{ij} = a_{ji}$ too. Therefore, S is symmetric.

To prove that S is semi-positive definite, we need to show that $X^T SX \geq 0$ for any vector $X = (x_1, x_2, \cdots, x_K)^T$ in R^K. This follows if we can prove that

$$\begin{aligned} X^T SX &= \sum_{all\ faces}\frac{(x_1 + x_2 + \cdots + x_{m_i})^2}{m_i} \\ &+ \sum_{all\ edges}\frac{(x_i + x_j)^2}{2} + \sum_{i=1}^{K}\frac{n_i^2 - 3n_i}{4}x_i^2 \\ &= Z_1 + Z_2 + Z_3 \end{aligned} \quad (6)$$

Note that from eq. (5) we immediately have the following expression for $X^T SX$:

$$\begin{aligned} X^T SX = \sum_{i=1}^{K}\Bigg[(\frac{n_i(n_i-1)}{4} + \sum_{j=1}^{n_i}\frac{1}{m_j})x_i^2 \\ + \sum_{j=1}^{n_i}(\frac{1}{2} + \frac{1}{m_j} + \frac{1}{m_{j-1}})x_{e_i(j)}x_i \\ + \sum_{j=1}^{n_i}\frac{\sum_{k=1}^{m_j-3} x_{f_{i,j}(k)}x_i}{m_j}\Bigg] \end{aligned}$$

where $e_i(j)$ is an indexing function for the edge points of x_i and $f_{i,j}(k)$ are indexing functions for the faces points of x_i. By re-arranging terms of the above expression, we have

$$\begin{aligned} X^T SX &= \sum_{i=1}^{K}\frac{n_i(n_i-1)}{4}x_i^2 + \sum_{i=1}^{K}\sum_{j=1}^{n_i}\frac{x_{e_i(j)}x_i}{2} \\ &+ \sum_{i=1}^{K}\sum_{j=1}^{n_i}\frac{\sum_{k=1}^{m_j} x_{f_{i,j}(k)}x_i}{m_j} \\ &= W_1 + W_2 + W_3 \end{aligned} \quad (7)$$

W_2 can be expressed as follows:

$$\begin{aligned} W_2 &= \sum_{i=1}^{K}\sum_{j=1}^{n_i}\frac{[(x_{e_i(j)} + x_i)^2 - x_{e_i(j)}^2 - x_i^2]}{4} \\ &= \sum_{i=1}^{K}\sum_{j=1}^{n_i}\frac{(x_{e_i(j)} + x_i)^2}{4} - \sum_{i=1}^{K}\frac{n_i}{2}x_i^2 \end{aligned}$$

Each edge is used twice in the first term on the right hand side of the above equation. Hence, we have

$$W_2 = \sum_{all\ edges}\frac{(x_j + x_i)^2}{2} - \sum_{i=1}^{K}\frac{n_i}{2}x_i^2$$

By substituting this expression into (7) for W_2, we get

$$X^T SX = \sum_{i=1}^{K}\frac{n_i(n_i-3)}{4}x_i^2 + \sum_{all\ edges}\frac{(x_j + x_i)^2}{2} + W_3$$

$$= Z_3 + Z_2 + W_3$$

where Z_3 and Z_2 are defined in (6). Z_1 in (6) can be expressed as

$$Z_1 = \sum_{all\ faces} \sum_{j=1}^{m_i} \frac{x_j(x_1 + x_2 + \cdots + x_{m_i})}{m_i}$$

The right side is nothing but W_3. Therefore, (6) is proved.

Next we prove that S is positive definite if the given matrix M has at least one vertex with a valence greater than 3. We prove this by contradiction. Without loss of generality we shall assume V_1 is a vertex of M with a valence greater than 3 and $X^T SX = 0$ for some $X \neq \mathbf{0}$.

Let x_i be an component of X that is not zero and let V_i be the corresponding vertex of M. Since M is connected, there exists a path in M: $(V_1,\ V_{j_1})$, $(V_{j_1},\ V_{j_2})$, ..., $(V_{j_p},\ V_i)$ that connects V_1 and V_i. Let x_1 and x_{j_k} be the corresponding components of V_1 and V_{j_k} in X, respectively. We see that x_1 must be equal to zero for, otherwise, we would have $X^T SX \geq n_1(n_1 - 3)x_1^2/4 > 0$, a contradiction. But then since

$$X^T SX \geq \frac{(x_1 + x_{j_1})^2}{2} + \frac{(x_{j_1} + x_{j_2})^2}{2} + \frac{(x_{j_2} + x_{j_3})^2}{2} + \cdots + \frac{(x_{j_p} + x_i)^2}{2}$$

we must have $x_{j_1} = 0$ for, otherwise, we would have $X^T SX \geq x_{j_1}^2/2 > 0$, a contradiction. By iterating this process, we would then have $x_{j_2} = 0$, $x_{j_3} = 0$, ..., and eventually $x_i = 0$. This contradicts to the assumption that $x_i \neq 0$. Hence, we must have $X^T SX > 0$ if $X \neq \mathbf{0}$ and this completes the proof that S is positive definite when M has at least one vertex with a valence bigger than 3.

As the product of two positive definite matrices D and S, the eigen values of A are positive. This follows immediately from the fact that DS and SD have the same eigen values (see, e.g., p.14 of [12]). On the other hand, since the sum of each rom of A is 1, we have $\|A\|_\infty = 1$. Hence, the eigen values of A are ≤ 1. This completes the proof.