# Locally Adjustable Interpolation for Meshes of Arbitrary Topology

Paper ID: 322

Category: Research Paper

The 3rd International Symposium on Visual Computing (ISCV07)

Lake Tahoe, Nevada/California, November 26-28, 2007

**Abstract.** Here is the abstract

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

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

# 1 Introduction

## 1.1 Previous Work

## 1.2 Overview

# 2 Basic Idea

For simplicity, we use $I(X)$ to represent the interpolation surface of mesh $X$, $S(X)$ to represent the limit surface of $X$ and $L(X)$ to represent all the limit points of $X$. For a given control mesh $M$, we need to find a smooth surface $I(M)$ that interpolates $M$. Suppose $S(M)$ is the limit surface of $M$ by using some subdivision scheme, say Catmull-Clark subdivision scheme. If we can find a surface $T_1$ or $K_1$, such that

$$T_1 + S(M) = I(M)$$

or

$$K_1 * S(M) = I(M)$$

then the interpolation problem is solved. Here $T_1$ (or $K_1$) can be regarded as an offset (scaling) surface which moves (scales) $S(M)$ to $I(M)$ evrywhere. We believe $T_1$ and $K_1$ are silimar to construct, hence it is sufficient to present one of them.

The difference of $I(M)$ and $S(M)$ at the vertices of $M$ can be calculated as follows.

$$M_1 = M - L(M).$$

Therefore $T_1 = I(M_1)$, i.e., $I(M_1)$ interpolates all the difference between $I(M)$ and $S(M)$. $M_1$ has the same topology as $M$, hence $I(M_1)$ and $I(M)$ are equally difficult to construct. However, the above process can be repeated to find a series of meshes $M_i$ $(1 \leq i \leq \infty)$ such that

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

and

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

Let $M = M_0$, from the above series we have

$$I(M) = \sum_{i=0}^{n} S(M_i) + I(M_{n+1}). \tag{2}$$

From eq. (1), we can get $M_i$ easily as follows.

$$M_i = (E - A)^i M_0, \tag{3}$$

where $E$ is the identity matrix and $A$ is the matrix that calculates all the limit points of the given mesh $M$. It is easy to see (the proof is shown in the appendix) that

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

Because $A$ is invertable (see the appendix), it also is easy to get

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

Combining the above two equations, we have

$$I(M) = S(\sum_{i=0}^{\infty} M_i) = S(A^{-1}M_0). \tag{4}$$

If we define

$$\sum_{i=0}^{\infty} M_i = \hat{M},$$

then $\hat{M} = A^{-1}M_0$ holds as well. Hence I(M) is also a subdivision surface and $\hat{M}$ is the mesh whose limit surface interpolates the given $M$. Traditionally, people try to directly find $A^{-1}M_0$ by solving an linear system [5, 12]. Hence it is difficult to deal with meshes of large number of vertices. However, with Eq. (4), $\hat{M}$ can be obtained by interatively applying eq. (1) until some given tolerance. Hence there is no problem to deal with large meshes. More importantly, just like Fourier transformation, any subdivision surface now can be represented by a summation of an infinite series of subdivision surfaces. For example, for any given mesh $M$, $S(M)$ can be represented with an infinite series of subdivision surfaces as follows.

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

Similar to Fourier transformation, we believe this good property can be used for a lot of applications in computer graphics and modelling, like fairing, smoothing, sharpening, lowpass or high pass filtering etc.

# 3 Test Results

The proposed techniques have been implemented in *C++* using *OpenGL* as the supporting graphics system on the Windows platform. Quite a few examples have been tested with the techniques described here. All the examples have extra-ordinary vertices. Some of the tested results are shown in Figures **??**, **??** and **??**. From these examples we can see smooth and visually pleasant interpolation shapes can be obtained.

# 4 Summary

Here is the Summary

# 5 Appendix

## 5.1 Proof of convergence of $(E - A)^i$

To prove this, we just need to show all the eigen values $\lambda_i$ of $A$ are $0 < \lambda_i \leq 1$. Here we present the proof using Catmull-Clark subdivision scheme. Other shcemes can be proven similarily. For Catmull-Clark subdivision scheme, the limit point of a vertex of degree $n$ is calculated as follows.

$$V_\infty = \frac{1}{n(n+5)}(n^2 V + \sum_{i=1}^{n} E_i + \sum_{i=1}^{n} F_i),$$

where $E_i$ and $F_i$ are the edge points and face points of vertex $V$, respectively. $A$ is a matrix satisfies:
1. $A_{ij} \geq 0$ and summation of each row is one, hence $\lambda_i \leq 1$;
2. A common coefficient $1/n_i/(n_i + 5)$ can be factored out for each row of $A$, where $n_i$ is the valance of vertex $i$ in the given mesh $M$. As a result, $A$ can be represented with the multiplicaiton of a diagonal matrix $diag(1/n_i/(n_i + 5))$ and a symetric matrix $B$. Hence $\lambda_i$ are always real numbers.

To finish the proof, we just need to show the eigen values of $B$ are bigger than 0, which is equavilant to prove $B$ is positive definite. This can be achieved by proving $X^TBX > 0$ for any vector $X \neq 0$. It is easy to see this if we expand $X^TBX$ as follows.

$$\begin{aligned} X^TBX \;\; = \;\; & \textstyle\sum_{allfaces}(x_i + x_j + x_k + x_r)^2 + \\ & 2 * \textstyle\sum_{alledges}(x_i + x_j)^2 + \\ & \textstyle\sum(n_i^2 - 3n_i)x_i^2 \end{aligned}$$

Because $n_i \geq 3$, hence $X^TBX > 0$ is always satisfied. In addition, we can see that $A$ is invertable because all its eigen values are bigger than 0.

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

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

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

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

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

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

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

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

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