Shape Control in Implicit Modeling

Baining Guo*
Department of Computer Science
Cornell University
Ithaca, New York 14853, USA

Abstract

Recently, implicit patches have emerged as alternative modeling primitives for three dimensional objects. In designing three dimensional models, one often encounters various shape requests. This paper develops techniques for satisfying such requests through shape control. In particular, we show how to achieve the convexity of quadric patches or cubic patches.

1 Introduction

The end goal of geometric modeling is to design and to manipulate three dimensional models represented by free-form surfaces. Traditionally, free-form surfaces are built from parametric patches. Parametric patches are successful as far as design and rendering are considered, but manipulating three dimensional models with parametric pathes poses fundamental difficulties. For example, parametric patches are not closed under sweeping and convolution. The intersection of two parametric patches are extremely difficult to represent and evaluate [HK86].

One way to avoid these problems is to build free-form surfaces from low-degree implicit patches. Implicit patches are closed under all common operations in geometric modeling [Baj88], and the intersections of low-degree implicit patches can be computed efficiently[OSS]. Recent research shows that quadric and cubic implicit patches are flexible enough for building arbitrary three dimensional models [Guo90, Guo91].

A major reason that parametric patches have become so popular in computer graphics is their good shape control properties. In this paper, we tackle the shape control issues of implicit patches. Using Bernstein-Bezier representation of polynomials, we can control the shapes of implicit patches through manipulating their control points.

*This work is supported by DARPA under ONR contract N00014-86K-0591, NSF Grant DMC-86-17355, and ONR Grant N00014-89J- 1946.

In designing free-form surfaces, one often encounters various shape requirements, such as a nice pattern of reflection lines and restrictions on the minimum radius of curvature. Among all the shape requirements, convexity is the most basic and the most frequently requested one. In this paper, we show how to manipulate the control points of a quadric patch or a cubic patch so that the patch become convex.

1.1 Previous work

Low-degree implicit surfaces are extensively used in the existing solid modeling and graphics systems as modeling primitives [RV83], and geometric operations on low-degree implicit surfaces are well understood [OSS]. Implicit surfaces are also very useful in surface fitting [PK89] and blending [RO87, MS85, HH87, Bli82].

Many authors have addressed the shape control of implicit patches [Sed85, WMW86, BW90]. In particular, Bloomenthal and Wyvill [BW90] discussed shape control using skeletons, and Sederberg pointed out that the Bernstein-Bezier representation are suitable for controlling the shapes of implicit patches [Sed85].

1.2 Overview

This paper is organized as follows. After giving some background information in Section 2, we describe the basic shape control techniques in Section 3. Section 4 shows how to achieve the convexity of quadric patches and cubic patches.

2 Bernstein-Bezier representation

Given a tetrahedron V with vertices $\mathbf{x}_1$, $\mathbf{x}_2$, $\mathbf{x}_3$, and $\mathbf{x}_4$, one can express any point $\mathbf{p}$ in space as:

$$\mathbf{p} = \sum_{i=1}^{4} \tau_i \mathbf{x}_i,$$

where

$$\sum_{i=1}^{4} \tau_i = 1.$$

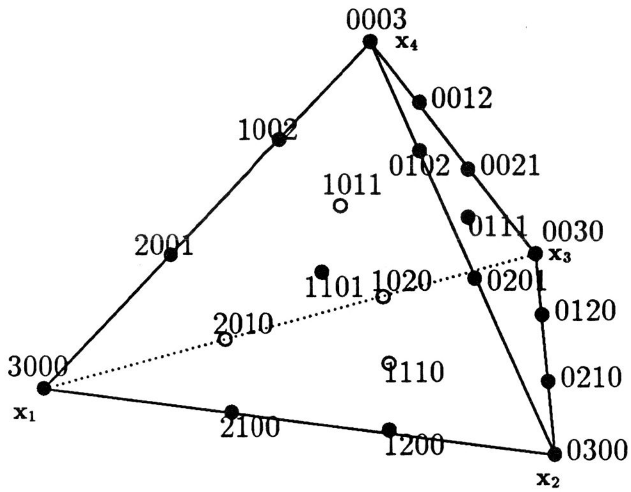

Figure 1: Cubic control points

The tuple $(\tau_1, \tau_2, \tau_3, \tau_4)$ is called the barycentric coordinate of $\mathbf{p}$. The barycentric coordinates are linearly related to Cartesian coordinates, so any implicit polynomial surface may be expressed in barycentric coordinates via a linear change of variables.

For a non-negative integer tuple $\lambda = (\lambda_1, \lambda_2, \lambda_3, \lambda_4)$ with $|\lambda| = \sum_{i=1}^{4} \lambda_i = n$, the Bernstein polynomial for λ is

$$B_\lambda^n(\tau) = \frac{n!}{\lambda_1!\lambda_2!\lambda_3!\lambda_4!}\tau_1^{\lambda_1}\tau_2^{\lambda_2}\tau_3^{\lambda_3}\tau_4^{\lambda_4}$$

Using Bernstein polynomials, one can uniquely represent any polynomial f of degree $\leq n$ as follows.

$$f(\tau) = \sum_{|\lambda|=n} b_\lambda B_\lambda^n(\tau)$$

The b_λ's are referred to as the *control points* of the polynomial f and its surface $S(f) = \{\mathbf{x}|f(\mathbf{x}) = 0\}$. The control points of a cubic polynomial are shown in Figure 1.

The following lemma is very useful.

Lemma 1 *If*

$$f(x) = \sum_{|\lambda|=k} c_\lambda B_\lambda^k(\tau)$$

and $c_{ke^i} = 0$, *then*

$$c_{(k-1)e^i+e^j} = (\nabla f(\mathbf{x}_i), \mathbf{x}_j - \mathbf{x}_i),$$

for $j = 1, 2, 3, 4$.

Proof: From [Dah86] ,

$$(\mathbf{x}_i - \mathbf{x}_j, \nabla f(\mathbf{x}))$$

$$= k \sum_{|\lambda|=k-1} (c_{\lambda+e^i} - c_{\lambda+e^j}) B_\lambda^{k-1}(\tau)$$

Letting $\mathbf{x} = \mathbf{x}_i$, we prove the lemma. ♣

3 The basic techniques for shape control

An implicit patch is defined as the zero contour of a polynomial f inside a tetrahedron $[\mathbf{x}_1\mathbf{x}_2\mathbf{x}_3\mathbf{x}_4]$[1].

$$f(\tau) = \sum_{|\lambda|=k} b_\lambda B_\lambda^k(\tau),$$

where τ is the barycentric coordinate defined by the tetrahedron $[\mathbf{x}_1\mathbf{x}_2\mathbf{x}_3\mathbf{x}_4]$. The basic idea of shape control is to express the geometric properties the implicit patch in terms of the control points so that one can achieve shape objectives by requiring the control points to satisfy certain constraints.

To get a feel for the effects of the control points, we study a univariate cubic polynomial f.

$$f(u) = b_{30}B_{30}^3 + b_{21}B_{21}^3 + b_{12}B_{12}^3 + b_{03}B_{03}^3.$$

The value of the function f over [0,1] and the convex hull of the points $(0, b_{30})$, $(1/3, b_{21})$, $(2/3, b_{12})$, and $(1, b_{03})$ are shown in Figure 2. The functions B_{30}^3, B_{21}^3, B_{12}^3, and B_{03}^3 are shown in Figure 3.

From these figures, we can see the following.

1. At the end points of the interval [0,1], the control points b_{30} and b_{03} equal to the function values of f.
2. The gradient of f at the end points of the interval [0,1] are determined by b_{30}, b_{21}, b_{12}, and b_{03}.
3. The control point b_{30} has a effect on the value of $f(u)$ for all u except $u = 0$, and the effect is the strongest near $u = 1$. Similar statement can be made about other control points.

All these relations between the control points and the properties of the polynomial f generalize to trivariate polynomials.

[1] We denote by $[\mathbf{x}_1, \cdots, \mathbf{x}_k]$ the convex hull of $\{ \mathbf{x}_1, ..., \mathbf{x}_k \}$

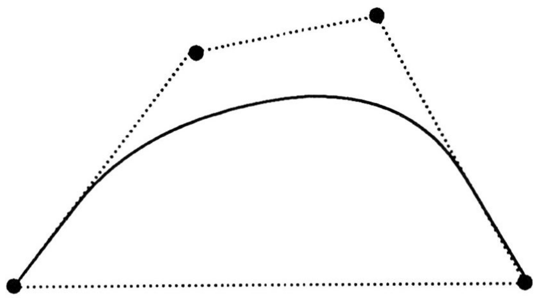

Figure 2: Function values of a univariate cubic function

Having understood the effect of the control points, we use the control points to control the shape of the surfaces. Consider the problem of interpolating points and lines in space by a surface $S(f)$. Since the values of f at a vertex of the tetrahedron $[\mathbf{x}_1\mathbf{x}_2\mathbf{x}_3\mathbf{x}_4]$ is equal to the value the control point at the vertex, setting the control point to zero forces the $S(f)$ to pass through the vertex. This method of interpolating a points can be generalized to a method of interpolating the edges of the tetrahedron $[\mathbf{x}_1\mathbf{x}_2\mathbf{x}_3\mathbf{x}_4]$.

Moving on to the problem of controlling the tangent plane of $S(f)$, we consider the tangent plane of $S(f)$ at the vertex $\mathbf{x}_1$. The tangent plane at $\mathbf{x}_1$ is defined by its the gradient $\nabla f(\mathbf{x}_1)$. From Section 2, we know that

$$b_{(k-1)e^1+e^j} = \frac{1}{k}(\nabla f(\mathbf{x}_1), \mathbf{x}_j - \mathbf{x}_1), \quad j = 2, 3, 4.$$

Since the vectors $\mathbf{x}_j - \mathbf{x}_1$ $(j = 2, 3, 4)$ are three linearly independent vectors, the above relation implies that the control points next to $\mathbf{x}_1$ completely determine the gradient $\nabla f(\mathbf{x}_1)$.

More sophisticated examples of shape control are easy to come by. The restriction of f to an edge of the tetrahedron $[\mathbf{x}_1\mathbf{x}_2\mathbf{x}_3\mathbf{x}_4]$ is a univariate polynomial. If the control points on the edge are all positive or all negative, then the surface $S(f)$ does not intersect the edge. Otherwise, the surface $S(f)$ intersects the edge exact once if there is exactly one sign change in the list of control points along the edge. Similar statements can be made for the faces of $[\mathbf{x}_1\mathbf{x}_2\mathbf{x}_3\mathbf{x}_4]$.

4 Achieving convexity

As an application of the techniques for shape control, we derive the convexity condition of an implicit patch in terms of its control points. Throughout this section, we concentrate on the implicit patch defined as the portion of a surface $S(f)$ inside a tetrahedron $V = [\mathbf{x}_1\mathbf{x}_2\mathbf{x}_3\mathbf{x}_4]$.

The reader is familiar with convex objects as a set of points in three dimensional space such that the line segment connecting two points in the set is contained in the set. Convex surfaces are often defined as surfaces whose Gaussian curvatures are positive over the entire surface. Convex objects and convex surfaces are related in that if a convex surface is closed and it bounds a point set with finite volume, then the point set is a convex object.

Defining a convex surface in terms of Gaussian curvature is not convenient when dealing with implicit surfaces. So we use the definition of convex surfaces in terms of the tangent planes. Let an implicit surface $S(f)$ have a tangent plane $S(P_{\mathbf{x}})$ at point $\mathbf{x} \in S(f)$. The surface $S(f)$ is convex at the point $\mathbf{x}$ if the surface $S(f)$ is in the half space bounded by $S(P_{\mathbf{x}})$ and pointed to by $-\nabla f(\mathbf{x})$. An implicit patch is convex if its primary surface is convex at every point on the implicit patch.

Notice the relationship between the convexity of the surface $S(f)$ and the convexity of the polynomial f. A polynomial f is convex over the tetrahedron V if for any two points $\mathbf{x}$ and $\mathbf{y}$ in the tetrahedron,

$$f(\frac{\mathbf{x}+\mathbf{y}}{2}) \leq \frac{1}{2}(f(\mathbf{x}) + f(\mathbf{y})).$$

It is easy to show that if the polynomial f is convex over the tetrahedron V, then the implicit patch defined as the portion of $S(f)$ inside V is convex. However, the converse is not true.

Motivated by the design of parametric convex surfaces, researchers in CAGD have obtained many results on the convexity conditions of polynomials over triangles [CP84]. It is possible to generalize these results to polynomials over tetrahedra [DM88], thus obtaining sufficient conditions for implicit patches to be convex. However, the convexity conditions obtained this way are often overly restrictive. So in the following, we derive the convexity conditions of an implicit patch directly from the definition of a convex implicit patch.

Let $\mathbf{p}' = (\tau_1', \tau_2', \tau_3', \tau_4')$ be a point close to a point $\mathbf{p} = (\tau_1, \tau_2, \tau_3, \tau_4)$ on the surface $S(f)$. The Taylor expansion of f, with higher order terms omitted, is

$$f(\mathbf{p}') = f(\mathbf{p}) + \sum_{i=1}^{4} \frac{\partial f}{\partial \tau_i}(\tau_i' - \tau_i) +$$

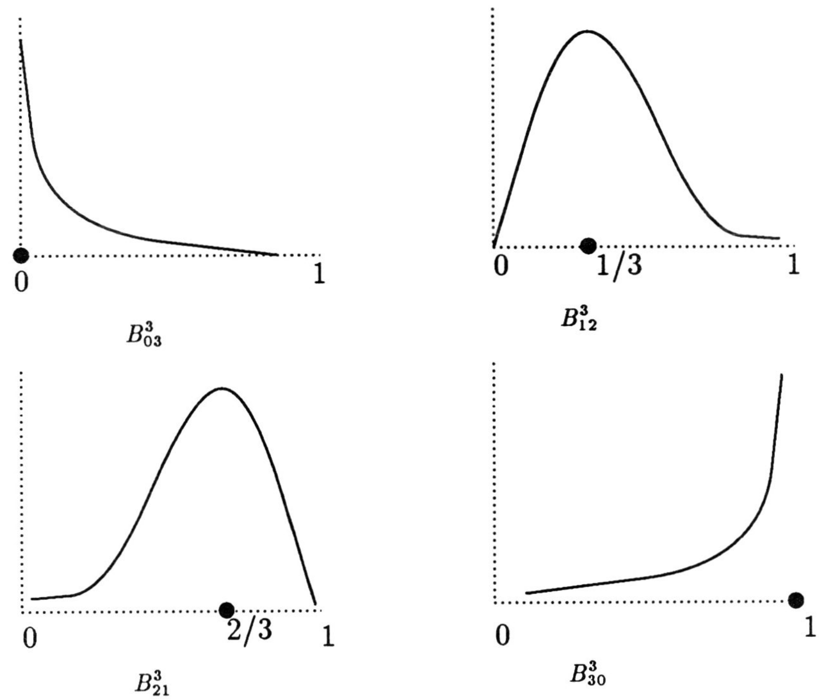

Figure 3: Weight functions for a univariate cubic function

$$\frac{1}{2}\sum_{i,j=1}^{4} \frac{\partial^2 f}{\partial \tau_i \partial \tau_j}(\tau_i' - \tau_i)(\tau_j' - \tau_j) \qquad (1)$$

The first term on the right-hand side of the above equation, $f(\mathbf{p})$, vanishes because $\mathbf{p}$ is on $S(f)$. Moving on to the second term, we notice that this term is the same as the left-hand side of the tangent plane equation at $\mathbf{p}$,

$$\sum_{i=1}^{4} \frac{\partial f}{\partial \tau_i}(\tau_i' - \tau_i) = 0. \qquad (2)$$

So the definition of a convex implicit surface implies that $S(f)$ is convex at $\mathbf{p}$ if

$$\sum_{i,j=1}^{4} \frac{\partial^2 f}{\partial \tau_i \tau_j}(\tau_i' - \tau_i)(\tau_i' - \tau_j) \geq 0 \qquad (3)$$

for all $\mathbf{p}'$. Introducing new variable $\sigma_i = \tau_i' - \tau_i$, we can write (3) as

$$\sum_{i,j=1}^{4} \frac{\partial^2 f}{\partial \tau_i \partial \tau_j}\sigma_i \sigma_j \geq 0. \qquad (4)$$

The σ's satisfy the constraint

$$\sum_{i=1}^{4} \sigma_i = 0 \qquad (5)$$

since the barycentric coordinates (τ_i) and (τ_i') satisfy the constraints

$$\sum_{i=1}^{4} \tau_i = 1 \text{ and } \sum_{i=1}^{4} \tau_i' = 1.$$

To eliminate the constraint (5), we substitute $-\sum_{i=1}^{3} \sigma_i$ for σ_4 in (4). The result is

$$\sum_{i,j=1}^{3} a_{ij}(\mathbf{p})\sigma_i \sigma_j \geq 0 \qquad (6)$$

for arbitrary $(\sigma_1, \sigma_2, \sigma_3)$ with

$$a_{ij}(\mathbf{p}) = \frac{\partial^2 f}{\partial \tau_i \partial \tau_j} + \frac{\partial^2 f}{\partial \tau_4 \partial \tau_4} - \frac{\partial^2 f}{\partial \tau_i \partial \tau_4} - \frac{\partial^2 f}{\partial \tau_j \partial \tau_4}. \qquad (7)$$

The condition (6) is the condition for the surface $S(f)$ to be convex at the point $\mathbf{p}$.

Applying the condition (6) to every point on an implicit patch, we have the following theorem.

Theorem 1 *An implicit patch is convex if the* 3×3 *matrix* $A = (a_{ij})$ *is positive definite for all points* $\mathbf{p}$ *on the implicit patch.*

Proof: Obvious from the above arguments. ♣

Generalizing the convexity conditions for bivariate polynomials would give a sufficient condition requiring the matrix A to be positive definite over the entire tetrahedron as opposed to the implicit patch. The condition in Theorem 1 is much less restrictive.

Although Theorem 1 gives a condition for the convexity of an implicit patch, the condition is hard to use because checking the condition for the infinitely many points on the implicit patch is impossible. So in the rest of the section, we use Theorem 1 to derive the convexity condition of an implicit patch in terms of its control points.

If f is a degree k polynomial given by

$$f = \sum_{|\lambda|=k} b_\lambda B_\lambda^k(\tau),$$

then

$$\frac{\partial^2 f}{\partial\tau_i\partial\tau_j} = k(k-1) \sum_{|\mu|=k-2} b_{\mu+e^i+e^j} B_\mu^{k-2}(\tau),$$

and $A = (a_{ij}(\mathbf{p}))$ is a 3×3 symmetric matrix whose entries are homogeneous degree $k-2$ polynomials in (τ_i). From linear algebra, A is positive definite if and only if

$$a_{11} \geq 0, \quad \begin{vmatrix} a_{11} a_{12} \\ a_{12} a_{22} \end{vmatrix} \geq 0, \text{ and } |A| \geq 0. \quad (8)$$

In order to decide whether A is positive definite for all points $\mathbf{p}$ on an implicit patch, we have to determine the signs of the minimum values of the quantities listed in (8) under the following the constraints,

$$f(\mathbf{p}) = 0, \quad (9)$$

$$\tau_1 + \tau_2 + \tau_3 + \tau_4 = 1, \text{ and } \tau_i \geq 0 \ (i = 1, 2, 3, 4). \quad (10)$$

Here the constraints characterize the points on the implicit patch. The inequality constraints and nonlinear constraints in (9) and (10) make the problem of deciding the convexity of a general implicit patch very hard.

Fortunately, practical criterions for the convexity of quadric patches and cubic patches can be derived. For quadric patches, notice that

$$\frac{\partial^2 f}{\partial\tau_i\partial\tau_j} = b_{e^i+e^j}$$

is independent of $\mathbf{p}$, so the convexity of a quadric patch can be decided by evaluating (8) with the constant $a_{ij} = b_{e^i+e^j} + b_{0002} - b_{e^i+e^4} - b_{e^j+e^4}$.

Deciding the convexity of a cubic patch is a little bit harder. Using the formula for Bernstein-Bezier polynomials, it is easy to verify that

$$\frac{\partial^2 f}{\partial\tau_i\partial\tau_j} = 6 \sum_{m=1}^{4} b_{e^m+e^i+e^j} \tau_m$$

Using this relation, we can rewrite (6) as

$$\sum_{m=1}^{4} \tau_m Q_m(\sigma) \geq 0. \quad (11)$$

where

$$Q_m(\sigma) = \sum_{i,j=1}^{4} (b_{e^i+e^j+e^m} + b_{2e^4+e^m} - b_{e^4+e^i+e^m} - b_{e^4+e^j+e^m})\sigma_i\sigma_j.$$

Since the left-hand side of inequality (11) is a convex combination of $Q_m(\sigma)$, the inequality (11) is valid over the entire tetrahedron enclosing the cubic patch if and only if the inequality is valid at the vertices of the tetrahedron, i.e.

$$Q_m(\sigma) \geq 0, \text{ for } m = 1, 2, 3, 4.$$

So the cubic patch is convex if the inequalities in (8) holds for $m = 1, 2, 3, 4$ with constant

$$a_{ij} = b_{e^i+e^j+e^m} + b_{2e^4+e^m} - b_{e^4+e^i+e^m} - b_{e^4+e^j+e^m}.$$

An important observation is that for each m, the above condition is exactly the same as the convexity condition for a quadric patch. Using the terminology of CAGD, we can say that a cubic patch inside a tetrahedron is convex if the subpolynomials at the vertices of the tetrahedron are convex.

Acknowledgement: I would like to thank Prof. J. Hopcroft for providing the support, guidance, and encouragement for this work.

References

[Baj88] C. Bajaj. Geometric modeling with algebraic surfaces. Technical Report CSD-TR-825, Com. Sci., Purdue University, 1988.

[Bli82] J. Blinn. A generalization of algebraic surface drawing. *ACM Transaction on Graphics*, 1:235–256, 1982.

[Bli84] J. Blinn. The algebraic properties of homogeneous second order surfaces. In *Siggraph84*, 84.

[BW90] J. Bloomenthal and B. Wyvill. Interactive techniques for implicit modeling. *Computer Graphics*, pages 109–116, 1990.

[CP84] G. Chang and P. Davis. The convexity of Bernstein polynomials over triangles. *Journal of Approximation Theory*, 40:11-28, 1984.

[DM88] W. Dahmen and C. Micchelli. Convexity and Bernstein polynomials over k-simploids. Technical Report RC13969, T. J. Watson Research Center, IBM, 1988.

[Dah86] W. Dahmen. Bernstein-Bezier representation of polynomial surfaces. *Extension of B-spline curve algorithms to surfaces*, SIGGRAPH'86 Course Notes, 1986.

[Guo90] B. Guo. Free-form surface modeling using implicit patches. *Proceedings of the Second Canadian Conference on Computational Geometry*, Ottawa, ON, 1990.

[Guo91] B. Guo. Surface generation using implicit cubics. to appear in *Proceedings of Computer Graphics International '91*, Cambridge, Massachusetts, June, 1991, Springer-Verlag.

[HH87] C. Hoffmann and J. Hopcroft. The potential method for blending surfaces and corners. In *Geometric Modeling: Algorithms and New Trends*, pages 347–366, 1987.

[HK86] J Hopcroft and D. Krafft. The challenge of robotics for computer science. In C. Yap and J. Schwartz, editors, *Advances in Robotics, Vol. 1: Algorithmic and Geometric Aspects of Robotics*, 1986.

[MS85] A. Middleditch and K. Sears. Blend surfaces for set volume modeling systems. *SIGGRAPH Computer Graphics*, 19:161–170, 1985.

[OSS] S. Ocken, J. Schwartz, and M. Sharir. Piecewise implementation of CAD primitives using rational parameterizations of standard surfaces. In *Planning, Geometry, and Complexity of Robot Motion*, pages 245–266, Norwood, NJ, 1983.

[PK89] N. Patrikalakis and G. Kriezis. Representation of piecewise continuous algebraic surfaces in terms of B-splines. *The Visual Computer*, 5:360–374, 1989.

[RO87] A. Rockwood and J. Owen. Blending surfaces in solid geometric modeling. In G. Farin, editor, *Geometric Modeling: Algorithms and New Trends*, 1987.

[RV83] A. Requicha and H. Voelcker. Solid modeling: current status and research directions. *IEEE Journal on Computer Graphics and Its Applications*, 3:25-37,1983.

[Sed85] T.W. Sederberg. Piecewise algebraic surface patches. *CAGD*, 2:53–59, 1985.

[WMW86] G. Wyvill, C. McPheeters, and B. Wyvill. Data structure for soft objects. *The Visual Computer*, 2, 1986.