SUPERHYPERBOLIC DISTANCE FUNCTIONS AND SPHERICAL CROSS PRODUCT FUNCTIONS: AN EXTENDED MODEL TO SUPERQUADRICS

Pi-Chung Hsu[1,2] *and Chungnan Lee*[2]

[1]Department of Information Management,
Fortune Institute of Technology, Kaohsiung County, Taiwan.
Email: d8834806@student.nsysu.edu.tw
[2]Department of Computer Science and Engineering,
National Sun-Yat-Sun University, Kaohsiung, Taiwan.
Email: cnlee@cse.nsysu.edu.tw

ABSTRACT

This paper presents two families of new functions for implicit surface modeling. The first family is superhyperbolic distance functions that can produce smooth deformable shapes bounded by some superhyperbolic surfaces, folded planes, square planes, or planes. The second family is spherical cross product functions that can generate an implicit sweep object. The latter can be viewed as an extended model to superquadrics, since not only parametric curves but also any implicit curve defined using a ray linear function can be used to perform a spherical cross product. Hence, it can generate a greater variety of shapes than superquadrics. Both functions have a number of advantages. They can be (1) parameterized, (2) further constructed to generate a complex object via set operations, (3) used as a field function for soft objects, and (4) applied to Blinn's blobby model and F-rep. Besides, some theorems are proposed to help users identify whether a function or a spherical cross product function is suitable as a field function.

Key words: Implicit surfaces; Soft objects; Superquadrics; Field functions; Sweep objects

1. INTRODUCTION

In implicit surface modeling, a complex object is defined by the blending or the set operations of some basic modeling primitives, such as superellipsoids, cylinders, and spheres, etc. To make implicit surface modeling more powerful, it is necessary to make the shapes in the basic modeling primitives more diverse. For this reason, a variety of basic modeling primitives have been proposed in the literatures, such as superquadrics [1], superquadric distance metrics [2], hyperquadrics [3], star-solid [4], generalized distance functions [5], and L_p distance metrics [6, 9]. Most of their shapes are superelliptic, bounded by a set of symmetrical and parallel plane-pairs via an intersection approximation operation [7].

To make these basic modeling primitive shapes even more diverse, we propose two new families of functions in this paper. The first family is superhyperbolic distance functions. Their shapes are bounded by a set of unbounded building blocks including superhyperbolic surface patches, folded planes, square planes, or parallel planes, via an intersection approximation operation. These functions can produce superelliptic, superhyperbolic, polyhedral, star-shaped, or rose-shaped objects.

The second family is spherical cross product functions that can generate an implicit sweep object and can be viewed as an extended model to Barr's superquadrics [1]. Superquadrics are created by performing spherical cross product on only two kinds of parametrically defined curves, superellipses and superhyperbolas. Instead of parametrically defined curves, any implicit curve defined using a 2D ray-linear function can be applied to the spherical cross product function for generating an implicit sweep object. For instance, we applied 2D superhyperbolic distance functions into the spherical cross product function, and developed a new family of functions, hyper-superquadrics.

Since hyper-superquadrics and superhyperbolic distance functions are inside-outside functions, they can be further constructed to generate an even more complex object via set operations. In addition, a user is allowed to adjust their parameter values to smoothly deform their shapes.

Since a field function must provide a means to calculate the influential scope [8], some theorems are proposed to help users identify whether a function is suitable as a field function, and to help users understand how to make the spherical cross product function suitable as a field function. We verify that superhyperbolic distance functions and hyper-superquadrics are suitable as field functions through these theorems.

To increase the rendering efficiency, we propose methods to parameterize superhyperbolic distance functions, spherical cross product functions, and hyper-superquadrics. In other words, the functions mentioned above have dual representations, in both implicit and parametric forms.

The remainder of this paper is organized as follows. Some basic knowledge about implicit surface modeling and soft objects is introduced in Section 2. Superhyperbolic distance functions and some theorems about ray-linear functions are discussed in Section 3. Section 4 presents spherical cross product functions, some related theorems, and the development of hyper-superquadrics. Conclusions are given in Section 5.

1.2 Notations

Before going to the next section, some notations are given as follows:

$\hat{X}$: Positions in 2D or 3D space.
$\vec{v}, \vec{u}$: Vectors which are interchangeable with $\hat{v}, \hat{u}$.
$R^+ : \{x \geq 0, x \in R\}$.
$f(\hat{X})$: Real-valued functions.
f^{-1} : The inverse of $f(\hat{X})$.
$f \circ g$: The composition of functions f and g.
$\vec{n}_i \bullet \hat{v}$: Inner product of $\vec{n}_i$ and $\hat{v}$.

2. PRELIMINARIES

An implicit surface can be defined as a point set $P = \{(x, y, z) \mid f(x, y, z) = c, (x, y, z) \in R^3\}$, where $f : R^3 \to R$ is called the defining function for the surface P, and $c \in R^+$. The defining function $f(x, y, z)$ can generate a scalar field for every position $\hat{p} = (x, y, z)$ in 3D space. The implicit surface P is an isosurface of the defining function f.

As stated in [6, 8, 9, 10], we know that field functions for soft objects are one kind of defining functions for implicit surfaces. As illustrated in Figure 1, a field function can be defined as

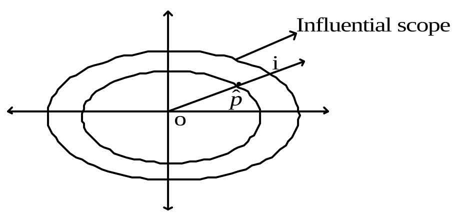

Figure 1. Influential scope of a field function.

$$F(x,y,z) = \begin{cases} f \circ d(x,y,z) & \\ 0 & r/R > 1, \end{cases} \tag{1}$$

$$d(x,y,z) = r/R = \|\overrightarrow{op}\| / \|\overrightarrow{oi}\| , \tag{2}$$

where f is called the potential function., $d(x, y, z)$ is called the distance function defined by an influential scope, called the front surface in [10], $r = (x^2 + y^2 + z^2)^{0.5}$, and $R = \|\overrightarrow{oi}\|$. Generally speaking, a influential scope must be a closed, continuous and star-shaped surface. The definition of a star-shaped surface is a surface that has at most one intersecting point with any vector from the origin. Since $F(z, y, z)$ is decreasing from 1 to 0, primitive soft objects $F_i(x, y, z) = 0.5$ can be smoothly blended by calculating the isosurface $\sum F_i(x, y, z) = 0.5$ without complex computations.

Many distance or defining functions have been developed to generate basic modeling primitives. They include:

(1). L_p distance metrics:

$$D_L(x, y, z) = ((|x/A|^{n_1} + |y/B|^{n_1} + |z/C|^{n_1})^{1/n_1} . \tag{3}$$

(2). Superquadrics distance metrics:

$$D_s(x, y, z) = ((|x/A|^{n_1} + |y/B|^{n_1})^{n_2/n_1} + |z/C|^{n_2})^{1/n_2} . \tag{4}$$

(3). Generalized distance functions [5]:

$$D_g(x, y, z) = (\sum_{i=1}^{r} |\vec{n}_i \bullet [x, y, z]|)^{1/n} . \tag{5}$$

For the ease of reading, "distance function" is used as the abbreviation for "distance function for soft object field function" in the following sections.

3. SUPERHYPERBOLIC DISTANCE FUNCTIONS

In contrast to the defining functions mentioned in Eqs. (3), (4), and (5), whose shapes are bounded by a set of symmetrical and parallel planes, this section introduces superhyperbolic distance functions, whose shapes are bounded by some superhyperbolic surface-pairs.

3.1 Superhyperbolic surface stripe functions

Let $\vec{v}$ and $\vec{u}$ be unit vectors in R^3, $\vec{v} \bullet \vec{u} = 0$, A, B >0, and $m > 0 \in R$. A superhyperbolic surface stripe function, denoted by $h(\hat{X}) : R^3 \to R^+$, can be written as

$$h(\hat{X}) = \begin{cases} ((m(\hat{X}))^m - (b(\hat{X}))^m)^{\frac{1}{m}} & \\ 0 \quad if \quad (m(\hat{X}))^m < (b(\hat{X}))^m , \end{cases} \tag{6}$$
$$m(\hat{X}) = \left|\hat{X} \bullet \vec{v}\right| \big/ A , \; b(\hat{X}) = \left|\hat{X} \bullet \vec{u}\right| \big/ B .$$

Geometrically, the shape of the inequality $h(\hat{X}) \leq c$ is a stripe bounded by two superhyperbolic , unbounded, and symmetrical surface branches $h(\hat{X}) = c$. $\vec{v}$ and $\vec{u}$ control the orientation of the surface branches. Figure 2 illustrates some possible 2D-shape changes when the parameter values in $h(\hat{X}) \leq c$ are changed as follows:

(1). When $m = 1$ and $c \leq 1$, it defines a stripe bounded by two folded planes, both of which pass through "points f, e, and g" and "points f' , e' , and g' ", respectively.

(2). When $m > 1$ and $c \leq 1$, it defines a stripe bounded by two superhyperbolic surface branches, both of which pass through "points a, e, and b" or "points a' , e' , and b' ", respectively.

(3). When $m = \infty$ and $c \leq 1$, it defines a stripe bounded by two square planes, both of which pass through "points a, c, e, d, and b" and "points a', c', e', d', and b' ", respectively.

(4). When $\mathrm{B} = \|\overrightarrow{\mathrm{oi}}\| = \infty$ and $c \leq 1$, it defines a stripe bounded by two near parallel planes, $m(\hat{X}) = 1$.

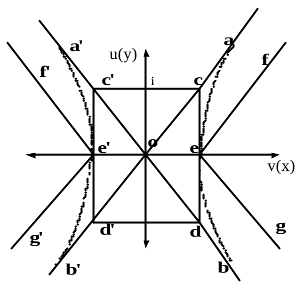

Figure 2. The shapes of the 2D superhyperbolic surface stripe functions. $A = \| \overrightarrow{oe} \|$ and $B = \| \overrightarrow{oi} \|$.

3.2 Basic superhyperbolic distance functions

Let V be a set of unit vector pairs, $V = \{(\vec{v}_a, \vec{u}_a) \mid \vec{v}_a \bullet \vec{u}_a = 0,\ \vec{v}_a, \vec{u}_a \in R^3\ , a = 1,2,\cdots,p\}$, $\hat{X} = (x, y, z)$ be a position in 3D space, and $n > 0 \in R$. Then every vector pair $(\vec{v}_a, \vec{u}_a) \in V$ can be used to define a superhyperbolic surface stripe function $h_a(\hat{X})$, as in Eq (6). One also can use all of them to define a basic superhyperbolic distance function, denoted by $H(\hat{X})$, as

$$V = \{(\overrightarrow{v_1}, \overrightarrow{u_1}), (\overrightarrow{v_2}, \overrightarrow{u_2}), \cdots, (\overrightarrow{v_p}, \overrightarrow{u_p})\},$$
$$H(\hat{X}) = \left(\sum_{a=1}^{p} \left(h_a(\hat{X})\right)^n\right)^{1/n}. \tag{7}$$

From Ricci's intersection approximation operation [7], the shape of the inequality $H(\hat{X}) \le c$ is bounded by the intersecting part of all of the superhyperbolic surface strips $h_a(\hat{X}) \le c, a = 1,2,\cdots,p$. In Figure 3, we used two different $h(\hat{X})$ to define a basic superhyperbolic distance function $H(\hat{X})$. Let $H(\hat{X}) = c$, it can be a closed or non-closed surface as shown in Figure 3.

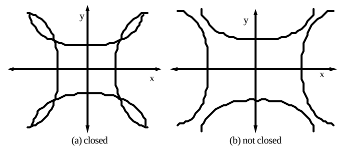

Figure 3. The shapes of the basic superhyperbolic distance functions.

If the surface, $H(\hat{X}) = c$, is closed, then it can produce an object bounded by a set of superhyperbolic surfaces, folded planes, square planes, or parallel planes.

3.3 Basic superhyperbolic distance functions for soft objects

In fact, not every closed implicit surface is suitable as the influential scope of a field function. As stated in [9], it depends on the complexity of the r/R calculation in Eq. (2). In order to help users identify whether a function is suitable as a distance function for point source field functions without worrying about the r/R calculation, some theorems are proposed. Through these theorems, superhyperbolic distance functions are proved to be distance functions for soft objects.

3.3.1 Ray-linear function versus distance function

Definition 3.1 Non-negative ray-linear function [5]

"Let V be a vector space. Then a function $f : V \to R^+$ is called non-negative ray-linear, if it satisfies the ray-linear property $f(a\hat{v}) = af(\hat{v})$ for any $\hat{v} \in V$ and $a \in R^+$."

For the ease of reading, "ray-linear function" is used to stand for "non-negative ray-linear function".

Theorem 3.2

"Let $f(\hat{X}): R^3 \to R^+$ be ray-linear and $f(\hat{X}) = c, c \in R^+$ be a closed and continuous surface, then $f(\hat{X}) = r/R$ is a distance function for a point source field function when the surface $f(\hat{X}) = 1$ is used as the influential scope."

From this theorem, one can directly use any ray-linear function as a distance function and need not to worry about the $d = r/R$ calculation and check if the influential scope is star-shaped. Hence, Eqs. (3), (4), and (5) are all distance functions because they are ray-linear.

3.3.2 Intersection of ray-linear functions

Performing an intersection operation on objects is very important to implicit surface modeling. So the following theorem is used to help identify whether the intersection approximation of ray-linear functions is ray-linear and is a distance function for soft objects.

Theorem 3.3

"Let $f_i(\hat{X}): R^3 \to R^+, i = 1,2,3,\ldots,p,$ be ray-linear, then $f(\hat{X}): R^3 \to R^+$, $f(\hat{X}) = \left(\sum_{i=1}^{p} (f_i(\hat{X}))^n\right)^{1/n}$, is a ray-linear function and also is a distance function."

3.3.3 Basic superhyperbolic distance functions for soft objects

Because every $h_a(\hat{X})$ in $H(\hat{X})$ is ray-linear, from Theorem 3.2 and 3.3 the basic superhyperbolic distance function $H(\hat{X})$ is a distance function for soft object if the surface $(H(\hat{X}))^n = 1$ or $H(\hat{X}) = 1$ is closed and used as the influential scope. Generally speaking, if the condition: "for any $\hat{X} \in R^3$, $\exists$ $h_a(\hat{X})$ in $H(\hat{X})$ such that

$((m_a(\hat{X}))^m - (b_a(\hat{X}))^m) > 0$ ” is satisfied, then $H(\hat{X}) = 1$ is closed because there exists at least one intersecting point between $h_a(\hat{X}) = 1$ and the vector $\hat{X}$.

3.4. Modified superhyperbolic distance functions

We also observed that the basic superhyperbolic distance function $H(\hat{X})$ in Eq. (7) has another modified form, denoted by $M_p(\hat{X})$, called the modified superhyperbolic distance function, which can be written as

$$M_p(\hat{X}) = ((M_{p-1}(\hat{X}))^{n_{p-1}} + \left(h_p(\hat{X})\right)^{n_{p-1}})^{1/n_{p-1}}, \qquad (8)$$
$$M_1(\hat{X}) = h_1(\hat{X}),\ \ n_1, n_2, \cdots, n_{p-1} > 0 \in R.$$

$M_p(\hat{X})$ is also a distance function because $M_p(\hat{X})$ is ray-linear.

Similar to $H(\hat{X}) = c$, geometrically $M_p(\hat{X}) = c$ can produce a surface bounded by the intersecting part of all of the $h_p(\hat{X}) \le c,\ p \in 1,2,\cdots,p$ in $M_p(\hat{X})$. The shape of $M_p(\hat{X}) = c$ is constructed from a set of superhyperbolic surfaces, parallel planes, folded planes, or square planes. The major difference between $H(\hat{X})$ and $M_p(\hat{X})$ is that $M_p(\hat{X})$ has more controlling parameters, from n_1 to n_{p-1}, to generate more deformed shapes than $H(\hat{X})$, which has only one parameter n, as in Eq. (7).

Here, some special shapes of $M_p(\hat{X}) = 0.5$ defined using the vector set $D = \{d_i \mid d_i = (v_i[x,y,z], A_i, m_i, u_i[x,y,z], B_i, n_{i-1}),\ i = 1,2,..., p\}$ are illustrated, where every $d_i \in D$ generates an $h_i(\hat{X})$, and n_0 is ignored. Let the set D for $M_5(\hat{X})$ be given as

{([1,0,0], 25, m_1,[0,1,0], 25,1), ([0,1,0], 25, m_2,[1,0,0], 25, n_1), ([$1/\sqrt{2}$,$1/\sqrt{2}$,0], 25, m_3,[-$1/\sqrt{2}$,$1/\sqrt{2}$,0], 25, n_2), ([-$1/\sqrt{2}$,$1/\sqrt{2}$,0], 25, m_4,[$1/\sqrt{2}$,$1/\sqrt{2}$,0], 25, n_3), ([1,0,0], 25,3, [0,0,0], 25, n_4) }, then some special shapes by changing the parameter m_i and n_i values in $M_5(\hat{X})$ are demonstrated as follows:

(1). Let $m_i = 2, i = 1,2,3,4,5$, and $n_4 = 2$, and let $n_1 = n_2 = n_3$ vary from 100, 16, 6, 2, 1, to 0.7, then it can generate objects from the top left to the bottom right in Figure 4(a). These objects are bounded by superhyperbolic surface-pairs.

(2). Let $m_i = 1, i = 1,2,3,4,5$, and $n_4 = 1$, and let $n_1 = n_2 = n_3$ vary from 100, 8, 2, 1.5, 1, to 0.7, then it can generate objects from the top left to the bottom right in Figure 4(b). These objects are bounded by folded plane-pairs. Thus, we know that $M_p(\hat{X})$ can produce star-shaped, rose-shaped, or polyhedral objects.

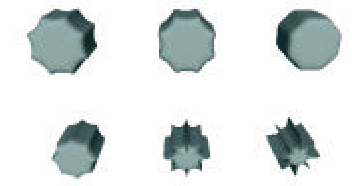

Figure 4(a). $m_i = 2, i = 1,2,3,4,5$, $n_4 = 2$, and $n_1 = n_2 = n_3$ vary from 100, 16, 6, 2, 1, to 0.7.

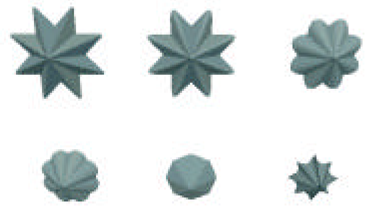

Figure 4(b). $m_i = 1, i = 1,2,3,4,5$, $n_4 = 1$, and $n_1 = n_2 = n_3$ vary from 100, 8, 2, 1.5, 1, to 0.7.

3.5 Parameterization of superhyperbolic distance functions

As stated in [5], the generation of parametrically defined shapes is fast since it is simply a polynomial computation. The parametric formulas for all of the functions introduced in Subsections 3.1-2, 3.4 now is presented. As stated by Akleman [5], a surface defined using a ray-linear function can be parameterized as

$$V(\boldsymbol{a}, \boldsymbol{b}, R) = g(\boldsymbol{a}, \boldsymbol{b})R + \hat{o}, \qquad (9)$$

where $g(\boldsymbol{a}, \boldsymbol{b})$ is an arbitrary unit vector staring from $\hat{o}$ to a point on the sphere $x^2 + y^2 + z^2 = 1$. R is the distance from $\hat{o}$ to the intersecting point between the vector $g(\boldsymbol{a}, \boldsymbol{b})$ and the parameterized surface. $\boldsymbol{a}$ and $\boldsymbol{b} \in [0,1]$ are the parameters in the sphere-coordinate.

From this idea, we determined that any star-shaped surface, $f(\hat{X}) = c$ or $(f(\hat{X}))^n = c$, where $f(\hat{X})$ is ray-linear, can be parameterized in the same way, because R can be solved by substituting $Rg(\boldsymbol{a}, \boldsymbol{b})$ into the parameterized equation. For instance, the surface $H(\hat{X}) = c$ in Eq. (7) can be parameterized, denoted by $V_b(\boldsymbol{a}, \boldsymbol{b})$, as follows:

$$H(\hat{X}) = c \quad \Rightarrow H(Rg(a,b)) = c \quad From\,\text{Eq.}\,(9)$$
$$\Rightarrow R = c / H(g(a,b)), \qquad (10)$$
$$\text{where } g(a,b) = \begin{bmatrix} \cos(2pa)\cos(pb - p/2) \\ \sin(2pa)\cos(pb - p/2) \\ \sin(pb - p/2) \end{bmatrix}.$$

Substituting Eq. (10) into Eq. (9) yields

$V_b(\boldsymbol{a},\boldsymbol{b}) = \left(c / H(g(\boldsymbol{a},\boldsymbol{b}))\right) g(\boldsymbol{a},\boldsymbol{b}) + \hat{o}$.

Similarly, the superhyperbolic surface $(H(\hat{X}))^n = c$ can be parameterized, denoted by $V_{b^n}(\boldsymbol{a},\boldsymbol{b})$, as $V_{b^n}(a,b) = (c^{1/n} / H(g(a,b)))g(a,b) + \hat{o}$.

Thus, the surfaces defined using modified superhyperbolic functions in Eq. (8) or superhyperbolic surface stripe functions in Eq. (6) can be parameterized in the same way.

4. SPHERICAL CROSS PRODUCT FUNCTIONS AND HYPER-SUPERQUADRICS

As stated in [1], superquadrics are defined by performing spherical cross product on two parametrically defined curves in 2D space. The spherical cross product $V_F(\boldsymbol{a},\boldsymbol{b})$ $= V_V(\boldsymbol{b}) \otimes V_H(\boldsymbol{a})$ of two curves, $V_V(\boldsymbol{b})$ and $V_H(\boldsymbol{a})$, can produce a surface $V_F(\boldsymbol{a},\boldsymbol{b})$ defined as

$$V_F(\boldsymbol{a},\boldsymbol{b}) = V_V(\boldsymbol{b}) \otimes V_H(\boldsymbol{a}) = \begin{bmatrix} t_x(\boldsymbol{a}) t_s(\boldsymbol{b}) \\ t_y(\boldsymbol{a}) t_s(\boldsymbol{b}) \\ t_z(\boldsymbol{b}) \end{bmatrix},$$

where $V_V(\boldsymbol{b}) = \begin{bmatrix} t_s(\boldsymbol{b}) \\ t_z(\boldsymbol{b}) \end{bmatrix}$ and $V_H(\boldsymbol{a}) = \begin{bmatrix} t_x(\boldsymbol{a}) \\ t_y(\boldsymbol{a}) \end{bmatrix}$ may be superellipses and superhyperbolas.

In fact, the spherical cross product can generate a variety of shapes, if curves other than superellipses and superhyperbolas are available. We therefore extend that idea to define a spherical cross product function based on two 2D functions instead of two parametrically defined curves. Not only superellipses and superhyperbolas, adopted by superquadrics, but also any other ray-linear functions can be applied into the spherical cross product function to generate a surface. Therefore, we developed a new family of functions, called hyper-superquadrics.

4.1 Spherical cross product functions

A spherical cross product function $F(x,y,z): R^3 \to R$ is written as

$$F(x,y,z) = V(s,z) \otimes H(x,y) = V(H(x,y),z) \quad (11a)$$

where $H(x,y)$ is ray-linear.

Simialr to superquadrics, the spherical cross product surface $F(x,y,z) = c$ can be parameterized as

$$V_F(a,b) = V_V(b) \otimes V_H(a) = \begin{bmatrix} t_x(a) t_s(b) \\ t_y(a) t_s(b) \\ t_z(b) \end{bmatrix}, \quad (11b)$$

where $V_V(\boldsymbol{b}) = \begin{bmatrix} t_s(\boldsymbol{b}) \\ t_z(\boldsymbol{b}) \end{bmatrix}$ and $V_H(\boldsymbol{a}) = \begin{bmatrix} t_x(\boldsymbol{a}) \\ t_y(\boldsymbol{a}) \end{bmatrix}$, $\boldsymbol{b} \in (-\boldsymbol{p}/2, \boldsymbol{p}/2)$, $\boldsymbol{a} \in [0, 2\boldsymbol{p})$, are the parametric formulas of $V(s,z) = c$ and $H(x,y) = 1$, respectively.

The spherical cross product function is based on two functions described below:

(1). Contour function $H(x,y): R^2 \to R^+$:

The equation $H(x,y) = 1$, called a contour curve, must be a continuous and closed curve centered at (0,0) in the x-y plane. In addition, the contour function $H(x,y)$ has to be ray-linear. The reasons will be explained in the next paragraph.

(2). Modulating function $V(s,z): R^2 \to R$:

The equation $V(s,z) = c$, called a modulating curve, must be a continuous curve in the x-z plane. Note that the variable s is viewed as the x-coordinate in the x-z plane.

Geometrically, the shape of the spherical cross product surface $V(s,z) \otimes H(x,y) = c$ is like the contour curve $H(x,y) = 1$ scaled (by s_a) and translated (by $[0,0,z_a]$) simultaneously by every point (s_a, z_a), $s_a \geq 0$, on the modulating curve $V(s,z) = c$. Since $H(x,y) \in R^+$, only the point $(s_a, z_a) \in V^{-1}(c)$, $s_a \geq 0$ can generate a curve $H(x,y) = s_a$, $z = z_a$ in 3D space (see Figure 5). In other words, $V(s,z) \otimes H(x,y) = c$ can generate a sweep object.

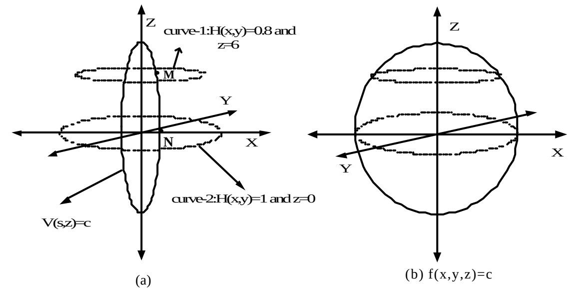

Figure 5. (a) The point $M(0.8, 6)$ on $V(s,z) = 1$ generates curve-1 $H(x,y) = 0.8, z = 6$, and the point $N(1, 0)$ on $V(s,z) = 1$ generates curve-2 $H(x,y) = 0.8, z = 0$. (b) Illustration of the real shape of $V(s,z) \otimes H(x,y) = 1$.

Making the profile on the x-z plane of the surface $V(s,z) \otimes H(x,y) = c$ similar to the shape of the modulating curve $V(s,z) = c$ is very important to geometric modeling. To attain this, the contour function $H(x,y)$ must be a ray-linear function. The reasons are:

(1). If $H(x,y)$ is ray-linear, then the curve $H(x,y) = s$ is like the contour curve $H(x,y) = 1$ scaled linearly by the value s from the modulating curve $V(s,z) = c$.

(2). The parametric formula $H^{-1}(s)$ of $H(x,y) = s$ is the parametric formula $H^{-1}(1)$ of $H(x,y)=1$ multiplied by s, that satisfies the parametric formula defined in Eq. (11b).

These reasons lead to the following Theorem 4.1.

Theorem 4.1

" $H(x,y): R^2 \to R^+$ is a ray-linear function if and only if $H^{-1}(s) = s\ H^{-1}(1)$ for any $s \in R^+$."

The spherical cross product function $V(s,z) \otimes H(x,y)$ can be more powerful, if it is an inside-outside function $V(s,z) \otimes H(x,y):\ R^3 \to R^+$. Thus, $V(s,z) \otimes H(x,y) = c$ can be further constructed to generate a complex object via set operations [7]. To attain this aim, one can let $V(s,z)$ map R^2 to R^+ and let curve $V(s,z)=c$ be closed on the positive side of the x-z plane.

To conveniently control the size of the surface $V(s,z) \otimes H(x,y) = c$, we introduce three alternatives:

(1). Alternative 1:

Let the size of the contour curve $H(x,y)=1$ and the modulating curve $V(s,z)=c$ both be about unit hypercircles individually. The spherical cross product surface $V(s,z) \otimes H(x,y) = c$ is scaled using an arbitrary vector $[a_x, b_y, c_z]$, yielding the following function by substituting $[x/a_x, y/b_y, z/c_z]$ into Eq. (11a):

$$F(x,y,z) = V(s, z/c_z) \otimes H(x/a_x, y/b_y), \qquad (12)$$

where $[a_x, b_y, c_z]$ is used to control the length along the x-axis, y-axis, and z-axis, respectively. In fact, Eq. (12) is equivalent to scaling the modulating curve by $[1, c_z]$ and to scaling the contour curve by $[a_x, b_y]$ simultaneously.

(2). Alternative 2:

Let the size of the contour curve $H(x,y)=1$ be about a unit hypercircle, $(x^n + y^n)^{1/n} = 1$, and the size of the modulating curve $V(s,z)=c$ be viewed as the exact size of the object's profile on the x-z plane. Its shape is then similar to a rotational sweep object generated using $V(s,z)=c$ as the profile to rotate around the z-axis along the curve $H(x,y)=1$.

(3). Alternative 3:

Let the size of the contour curve $H(x,y)=1$ be the exact size of the object's profile on the x-y plane, and let the height of the modulating curve $V(s,z)=c$ be the exact height of the object. To do this, the s-axis length of $V(s,z)=c$ must be normalized or scaled to be 1. If the s-axis length of $V(s,z)=c$ is $n > 0 \in R$, then the s-axis of $V(s,z)=c$ must be scaled by $1/n$, in order to make the s-axis length 1. Similar to Eq. (12), it thus yields

$$F(x,y,z) = V(ns,z) \otimes H(x,y) = V(nH(x,y), z)$$
$$= V(H(nx,ny), z) = V(s,z) \otimes H(nx,ny). \qquad (13)$$

From the discussions above, we conclude that any ray-linear function can be used as the contour function and the modulating function in Eq. (11a), (12), or (13), to produce a 3D closed surface. Generally speaking, not only superellipses and superhyperbolas, which are adopted by superquadrics, but also $D_L(x,y)$ in Eq. (3), $D_g(x,y)$ in Eq. (5), and superhyperbolic distance functions $H(x,y)$ in Eq. (7) and $M_p(x,y)$ in Eq. (8) can be used as the contour and modulating functions since they are all ray-linear. Hence, the spherical cross product function is the generalized form of superquadrics. Since the modulating function does not have to be ray-linear, $(D_L(x,y))^n$, $(D_g(x,y))^n$, $(M_p(x,y))^n$, and 2D hyper-quadrics [3] also can be used as a modulating function because they can produce a closed curve.

4.2 Hyper-superquadrics

As mentioned in Section 4.1, the 2D Modified superhyperbolic distance functions, $M_p(\hat{X}): R^2 \to R^+$ in Eq. (8), can be used as the contour and the modulating functions in Eqs. (11a), (12), and (13), respectively. Consequently, a new family of functions called hyper-superquadrics is developed. They can be written as follows:

$$F(x,y,z) = M_p(s,z) \otimes M_p(x,y), \qquad (14a)$$

$$F(x,y,z) = M_p(s, z/c_z) \otimes M_p(x/a_x, y/b_y), \qquad (14b)$$

$$F(x,y,z) = M_p(s,z) \otimes M_p(nx,ny). \qquad (14c)$$

4.2.1 Hyper-superellipsoids

If $b_a(\hat{X})$ in every $h_a(\hat{X}), a = 1,2,\cdots,p$, of $M_p(\hat{X})$ in Eq. (6), are omitted from Eqs. (14a), (14b), and (14c), then it generates a family of functions called hyper-superellipsoids. Thus $h_a(\hat{X}), a = 1,2,\cdots,p$ can be written as $h_a(\hat{X}) = m_a(\hat{X}) = |\hat{X} \bullet \vec{v}_a| / A$.

Then the shapes $M_p(x,y)=1$ and $M_p(s,z)=c$ both are constructed from some parallel line-pairs $m_a(\hat{X})$. So hyper-superellipsoids are hyper-superelliptic and have smoothing polyhedral profiles in the x-y or the x-z planes. Some hyper-superellipsoids are demonstrated in Figure 6. Let the set D of $M_4(x,y)$ be {([1,0],1,1,[0,0],1,1), ([0,1],1,1,[0,0],1, n_1), ([$1/\sqrt{2}$, $1/\sqrt{2}$],1,1, [0,0],1, n_2), ([-$1/\sqrt{2}$, $1/\sqrt{2}$],1,1,[0,0],1, n_3)}, and the set D of $M_4(s,z)$ be {([1,0],1,1,[0,0],1,1), ([0,1],1,1,[0,0],1, n_1), ([$1/\sqrt{2}$, $1/\sqrt{2}$],1,1,[0,0],1, n_2), ([-$1/\sqrt{2}$, $1/\sqrt{2}$],1,1,[0,0],1, n_3)}.

From the top left to the bottom right in Figure 6, the first four surfaces show the shape changes of $M_p(s, z/30) \otimes M_p(x/30, y/30) = 0.5$ as $n_i, i = 1,2,3$ in $M_4(s, z)$ are set to 100 and $n_i, i = 1,2,3$ in $M_4(x, y)$ vary from 100,10,6, to 2; the last two surfaces show the shape changes of $M_p(s, z/30) \otimes M_p(x/30, y/30) = 0.5$ as $n_i, i = 1,2,3$ in $M_4(s, z)$ vary from 6 to 2 and $n_i, i = 1,2,3$ in $M_4(x, y)$ are set to 6.

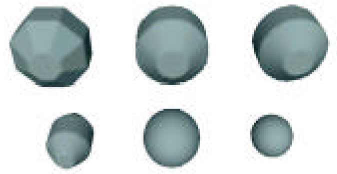

Figure 6. Hyper-superellipsoids.

4.2.2 Hyper-superhyperboloids

If every $h_a(\hat{X})$, $a = 1,2,\cdots, p$ in $M_p(\hat{X})$, in Eqs. (14a), (14b), and (14c), is defined as

$$\left((m_a(\hat{X}))^m - (b_a(\hat{X}))^m\right)^{1/m},$$

in Eq. (6), then it generates hyper-superhyperboloids. As mentioned in Subsection 3.1, every $h(\hat{X}) = c$ can produce a unbounded superhyperbolic curve, folded lines, square lines, or parallel lines. Users can view every $h(\hat{X})$ in $M_p(\hat{X})$ as a building block to construct their contour and modulating curves freely. This characteristic is very useful, especially when it is implemented in interactive graphics.

Figure 7(a) shows some special contour and modulating curves defined using $M_p(\hat{X})$, where folded-lines can be defined using $h(\hat{X})$ and straight lines can be defined using $m(\hat{X})$. They can be further deformed by adjusting the values of parameters m and n in $M_p(\hat{X})$.

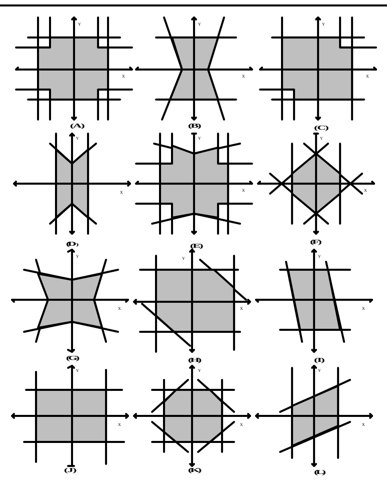

Figure 7(a). Contour Curves or modulating curves, each of which can be defined using $M_p(\hat{X})$.

Figure 7(b) shows the objects defined using the curves from the first three rows in Figure 7(a) as the modulating curves, where the curve (k) in Figure 7(a) is used as the contour curve.

Figure 7(b). Hyper-superhyperboloids, where the curve (K) in Figure 7(a) is used as the contour curves.

4.3. Spherical cross product distance functions

In this section, some theorems are proposed to help the user identify whether the spherical cross product function is a distance function.

Theorem 4.2

"Let $F(x, y, z) = V(s, z) \otimes H(x, y)$ be a spherical cross product function, then it is a ray-linear function if the $V(s, z)$ is a ray-linear function."

Theorem 4.3

"Let $F(x, y, z) = V(s, z) \otimes H(x, y)$ be a spherical cross product function. Let $V(s, z): R^2 \to R^+$ be ray-linear, then $F(x, y, z)$ can be a distance function of the point-source field function for soft objects if the surface $F(x, y, z) = 1$ is used as the influential scope and $V(s, z) = 1$ is closed on the positive side of the x-z plane".

From Theorems 4.2 and 4.3, one can conclude that hyper-superquadrics are distance functions for point-source field functions, because $M_p(s, z)$ and $M_p(x, y)$ in Eqs. (14a), (14b), and (14c) are ray-linear.

4.4 Parameterization of hyper-superquadrics

Here we present two methods to parameterize hyper-superquadrics as follows:

(1). Since hyper-superquadrics are ray-linear, one can use the method stated in Subsection 3.5 to parameterize them.

(2). Since the modulating curve $V(s, z) = c$ and the contour curve $H(x, y) = 1$ are defined using a ray-linear function $M_p(\hat{X})$, they can be parameterized, respectively, using the method stated in Subsection 3.5, and then substitute them into Eq. (11b) to generate the parametric formula.

5. CONCLUSIONS

One of the problems of implicit surface modeling is that basic modeling primitives are not diverse enough. To cope with this problem, we have developed two families of functions as follows:

(1). Superhyperbolic distance functions:

These functions can produce objects bounded by a set of square planes, folded-planes, superhyperbolic surfaces, or planes.

(2). Spherical cross product functions:

These functions can produce a sweep object in implicit and parametric forms. From Theorem 4.1, one can freely define two ray-linear functions, and then substitute them into the spherical cross product function to generate an implicit sweep soft object.

In addition, these two families of functions have the following important properties:

(1). They can be deformed smoothly by adjusting parameter values.

(2). Each of them can be used directly as a distance function for soft object field functions.

(3). They can be applied to F-rep [11] and Blinn's blobby model [12].

(4). They can be further constructed to create a complex object via blending or set operations

(5). They have dual representations, in both implicit and parametric forms.

Besides, we have proposed Theorems 3.2-4 to help users easily identify and discover a new distance function for soft objects without having to worry about the r/R calculation. Through Theorems 4.1-3, one can define new ray-linear functions, and then use them as the contour and modulating functions for spherical cross product functions to generate a variety of implicit sweep soft objects.

6. REFERENCES

[1] H. Barr, "Superquadrics", IEEE Computer Graphics and Applications, 1(1), pp. 11-23, 1981.

[2] M. Tiggers, M. S. T. Carpendale, and B. Wyvill, "Generalized distance metrics in implicit surface modeling", Proceedings of the Tenth Western Computer Graphics Symposium, pp. 14-18, Banff, AB, March 1999.

[3] A. Hanson, "Hyperquadrics: smoothly deformable shapes with convex polyhedral bounds", Computer Vision, Graphics and Images Processing, 44(1), pp. 191-210, 1988.

[4] E. Akleman, "Interactive construction of smoothly blended star solids", Proceedings of Graphical Interface' 96, May 1996.

[5] E. Akleman and J. Chen, "Generalized distance functions", Proceedings of Shape Modeling International' 99, pp. 72-79, 1999.

[6] C. Blanc and C. Schlick, "Extended field functions for soft objects", Implicit surfaces' 95, pp. 21-32, April 1995.

[7] A. Ricii, " A Constructive Geometry for Computer Graphics", The Computer Journal, 16(2), pp. 157-160, May 1973.

[8] G. Wyvill, C. McPheeters, and B. Wyvill, "Data structure for soft objects", The Visual Computer, 2(4), pp. 227-234, 1986.

[9] G. Wyvill and B. Wyvill, "Field functions for implicit surfaces", The Visual Computer, vol. 5, pp. 78-52, 1989.

[10] B. Crespin, C. Blanc, and C. Schlick, "Implicit sweep objects", EUROGRAPHICS' 96, 15(3), pp. 165-175, 1996.

[11] A. Pasko, V. Adzhiev, A. Sourin, V. Savchenko, "Function representation in geometrical modeling: concepts, implementation and applications", The Visual Computer 11(8), pp. 429-446, 1996.

[12] J. F. Blinn, "A generalization of algebraic surface drawing", ACM transactions on graphics, 1(3), pp. 235-256, 1982.