

International Journal of Shape Modeling  
© World Scientific Publishing Company

## CONTROLLABLE BINARY CSG OPERATORS FOR “SOFT OBJECTS”

LOÏC BARTHE

*SIRV Group, IRIT/UPS Toulouse  
118 route de Narbonne, 31062 Toulouse Cedex 4, France  
lbarthe@irit.fr  
<http://www.irit.fr/Loic.Barthe/>*

BRIAN WYVILL

ERWIN DE GROOT

*Dept. of Computer Science, University of Calgary  
2500 University Dr. NW Calgary, Alberta, T2N 1N4, Canada  
blob@cpsc.ucalgary.ca  
erwin@erwindegroot.nl  
<http://pages.cpsc.ucalgary.ca/blob/>*

Received (Day Month Year)

Revised (Day Month Year)

Accepted (Day Month Year)

Communicated by (xxxxxxxxxx)

Potential functions allow the definition of both an implicit surface and its volume. In this representation, two categories can be distinguished: bounded and unbounded representations. Boolean composition operators are standard modelling tools allowing complex objects to be built by the combination of simple volume primitives. Though they are well defined for the second category, there is no clear definition of the properties that such operators should satisfy in order to provide bounded representation with both smooth and sharp transition. In this paper, we focus on bounded implicit representation. We first present fundamental properties to create adequate composition operators. From this theoretical framework, we derive a set of Boolean operators providing union, intersection and difference with or without smooth transition. Our new operators integrate accurate point-by-point control of smooth transitions and they generate  $G^1$  continuous potential fields even when sharp transition operators are used.

*Keywords:* Implicit modelling, Soft objects, Bounded representation, CSG operators, Blending.

1991 Mathematics Subject Classification: 22E46, 53C35, 57S20

### 1. Introduction

Providing interactive, accurate and intuitive control of shapes is a fundamental issue in the development of three-dimensional modelling techniques. Direct manipulation of meshes, parametric shape representations and, more recently, subdivision surfaces are common and useful solutions adopted by most commercial software. However, implicit volume models<sup>1,2</sup> are rapidly becoming a practical alternative to these methods due to the increase

2 *Loïc Barthe and Brian Wyvill and Erwin de Groot*

in computer power and storage capacity of modern workstations combined with the latest developments in graphics hardware.

Among the advantages of implicit surfaces we notice their natural blending property<sup>3,4,5</sup>, the true three-dimensional representation of volumes, the efficiency for collision tests<sup>2</sup>, the Boolean composition of their volume by simple function composition<sup>6,8,9,10</sup> and finally their very compact functional representation.

Two general implicit representations can be distinguished: A bounded representation where the function defining the volume returns a constant value outside the boundary (known as “Metaballs”<sup>11</sup> or “Soft Objects”<sup>12</sup>), and an unbounded representation (such as  $R$ -functions<sup>8,9</sup>) where the function varies in the whole space. Unbounded representation provides a general implicit volume representation<sup>13</sup> and therefore, a wide variety of modelling operators such as sweeping by moving solids<sup>14</sup>, Boolean composition with soft transition<sup>9,15,16,17</sup> and Constructive Volume Geometry algebra<sup>18</sup>, have been proposed. Due to their global definition, it is difficult to provide both accurate and interactive surface rendering<sup>30</sup>. The local representation of bounded objects is better suited for this purpose. “Soft Objects” are more popular for their automatic blending property and whereas many blending functions have been presented in the literature (see<sup>20,21,22</sup> for a summary), as far as we know, since Ricci<sup>6</sup>, no consequent improvement has been proposed for Boolean composition operators on “Soft Objects”. Under its actual form, composition with sharp transition generates undesirable discontinuities in the potential field and smooth composition provides a very limited control of the form of the transition.

Smooth transitions in the composition operators have become a standard tool for implicit modelling. Intuitive parameters for control are necessary in order to allow the user to design the desired shape. On the other hand it has been shown that the local definition of the bounded representation is a fundamental advantage which provides complete interactive modelling tools<sup>23,24</sup>. Therefore providing Boolean composition operators (with or without smooth transition) for bounded implicit primitives with properties well suited for CSG compositions remains an open, rather important problem.

In this paper, we first state the limits of the actual methods used to combine bounded objects. We then present specific properties that composition operators should satisfy in order to provide the surface resulting from a composition with, at least, a  $G^1$  continuous potential field. We introduce a generic family of composition functions based on arc-of-ellipses. From this general function representation, we derive new operators to combine bounded primitives with both smooth transition integrating “point-by-point” control of the shape (as introduced by Barthe et al<sup>25</sup>), and sharp transition with a  $G^1$  continuous potential field everywhere outside the surface and the boundary.

## 2. Related works

“Metaballs”<sup>11</sup> or “Soft Objects”<sup>12</sup> are bounded objects defined by a potential function  $f : \mathbb{R}^3 \rightarrow \mathbb{R}$ . A single primitive, also called “skeleton primitive”, is first defined from a simple geometric object  $S$  (the skeleton) such as a point, a line, a polygon, etc. Then one has to choose a metric  $d$  which is generally the Euclidean norm. The potential function  $f$

is defined as a function of the distance with respect to the norm  $d$  from the skeleton  $S$  to points  $p$  of  $R^3$ :

$$\begin{aligned} f : R^3 &\rightarrow R \\ p &\rightarrow f(r) \quad \text{with } r = d(S, p). \end{aligned} \quad (1)$$

We denote  $r$  as  $d(S, p)$ . The primitive’s boundary is defined by a chosen scalar  $R$  called “radius of influence”. The function  $f$  equals 0 if  $r \geq R$  and it decreases from 1 to 0 when  $r$  increases from 0 to  $R$ , following a Gaussian-like variation. The surface is defined by the set of points  $p_0 \in R^3$  such that  $d(S, p_0) = r_0$  and  $f(r_0) = C$ , where  $C$  is a pre-chosen value in  $(0, 1)$  (usually  $1/2$ ), and the volume object is defined by the set of points of  $R^3$  for which  $f(r) \geq C$  (see Figure 1). A wide variety of primitives are available<sup>4,26</sup>, and the blend of a set of  $n$  primitives is automatically computed by summing their potential functions  $f_i$  ( $i = 1..n$ ):

$$F(f_i) = \sum_{i=1}^n f_i, \quad (2)$$

where  $F$  is a new potential function which has the same analytic properties as functions  $f_i$ . Many different field functions<sup>20,21,22</sup> and blending models<sup>27,28</sup> have been proposed to control the smoothness of the transition region, but the operators remain limited to the blending and the control of which primitives must and must not blend. The locality of the definition and the capacity to be automatically blended allow modelling techniques based on these objects to be interactive<sup>24,29</sup>.

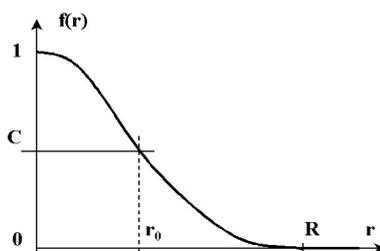


Fig. 1. Graph of a potential function  $f$  defining a “Soft Object”.

CSG composition operators are already supported by bounded primitives (using the Ricci’s min/max operators<sup>6</sup>) but discontinuities are introduced in the gradient of the potential field of the resulting object, altering the smoothness of the transition when it has to be blended (Figure 2). This is undesirable.

A solution using Ricci’s super-elliptic operator<sup>6</sup> to apply binary union and binary intersection operators with smooth transitions to “Soft Objects” was used by Wyvill et al<sup>23</sup> (see Equation 3), and extended to  $n$ -ary operations.

$$G(f_1, f_2) = (f_1^n + f_2^n)^{\frac{1}{n}}. \quad (3)$$

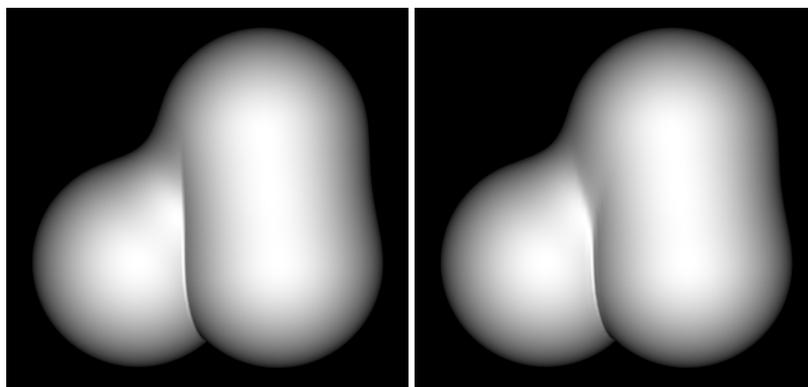
4 *Loïc Barthe and Brian Wyvill and Erwin de Groot*

Fig. 2. On the left, the two bottom spheres are first combined (union) with sharp transition using Ricci's max operator. The resulting object is then combined with the top sphere (union) using smooth transition. We can see the undesirable discontinuity in the middle of the smooth transition. On the right we show a correct smooth transition.

This operator has a single parameter  $n$  which controls the transition sharpness. There is no explicit link with some geometric parameter allowing the user to interactively select the beginning, the end, or the form of the transition. The user is limited to approximately select either, the global sharpness of the transition, or where the transition starts on one of the combined primitives, or where it finishes on the other one (in the case of a binary operator).

Recently Barthe et al.<sup>30</sup> and Hsu et al.<sup>17</sup> introduced new CSG operators with smooth transitions. In Hsu et al., operators can be  $n$ -ary and a lot of flexibility is provided in the choice of the function defining the operator. However, the control over the shape remains limited and the operator is very expensive to evaluate. On the other hand, Barthe et al. proposed binary operators based on<sup>16</sup> which are cheaper to evaluate and which provide a high level of control on the shape of the transition. Since our goal is to allow the user to accurately control the shape of the smooth transition, we use operators presented in<sup>16</sup> as a basis to develop our operators for "soft objects".

### 3. Fundamental properties

As far as we know, there is no clear definition of what is to be expected from a CSG Boolean composition operator on bounded implicit primitives. However, some first insights are given by the properties satisfied by the composition operators on unbounded primitives<sup>5,25</sup>. The main difference between the *two* representations is the boundary. While we are expecting equivalent properties in terms of potential field variations and shape control, we also have to maintain a consistent potential field around and at boundaries. The constraints can be specified as follows:

- The potential field produced by the composition operator must be at least  $G^1$

continuous everywhere inside the object boundary.

- The “automatic blending” property by potential summation must be conserved through the composition.
- The result of a composition must be a bounded object having its bounding box easily computed from those of the combined primitives.
- For smooth composition, the extremities of the transition should be able to be intuitively selected once the objects’ boundaries (radii of influence) are fixed.

In order to introduce these properties, we first look at the expected result and work backwards, up to the composition operator. We only present the union Boolean operator because once this case is understood, properties for intersection and difference can be directly derived without major difficulty.

Figure 3 illustrates the result of the union of two spheres with smooth transition. Object  $i$  ( $i = 1, 2$ ) is defined by a potential function  $f_i$ . The equation  $f_i = C$  represents object’s  $i$  surface, and inequalities  $f_i > C$  and  $f_i < C$  define the inside and outside of object  $i$  respectively. Equation  $f_i = 0$  defines the part of space  $\mathbb{R}^3$  on and outside object’s  $i$  boundary (zone 4 in Figure 3).

From the graph shown in Figure 3 we plot the union binary combination operator  $G$  (Figure 4) and we deduce the definition of operator  $G$  in each zone:

- In zone 1,  $f_1 > 0$  and  $\forall f_1, f_2 = 0$ . Therefore, in this area, operator  $G$  is a *one-dimensional* map  $G(f_1, 0)$  which scales the values of function  $f_1$ .
- In zone 2,  $f_2 > 0$  and  $\forall f_2, f_1 = 0$ . Therefore, in this area, operator  $G$  is a *one-dimensional* map  $G(0, f_2)$  which scales the values of function  $f_2$ .
- In zone 3,  $f_1 > 0$  and  $f_2 > 0$ . Operator  $G$  is a *two-dimensional* function  $G(f_1, f_2)$  which defines a *two-dimensional* potential field.
- In zone 4,  $f_1 = f_2 = 0$ . Here,  $G(f_1, f_2) = G(0, 0) = C^e, C^e \in \mathbb{R}$ .

Function  $G$  can define a *two-dimensional* potential field only in zone 3. Hence, the transition between the combined primitives has to be fully defined in this zone, and no transition can be performed outside one of the objects boundary. This also implies that the continuity between the transition and the combined primitives has to be ensured in this zone, or at its boundary (as shown with small circles in Figure 4). Since we want to produce smooth potential fields, the continuity at the junction has to be at least  $G^1$  i.e. the partial derivatives of function  $G$  must satisfy the following properties:  $\partial G / \partial f_2 = 0$  at the junction between zones 1 and 3, and  $\partial G / \partial f_1 = 0$  at the junction between zones 2 and 3.

In zone 1, function  $G$  scales the values of  $f_1$  and if  $G(f_1, 0) = f_1$ , operator  $G$  reproduces the metric and the variations of function  $f_1$ , and the potential field defined by  $f_1$  is preserved through composition. This property avoids the introduction of non-uniform variations in the potential fields which could alter the regularity of the transition and the “automatic blending” property. In zone 2, function  $G$  scales the value of  $f_2$  and as argued for zone 1, a pertinent definition of operator  $G$  is:  $G(0, f_2) = f_2$ .

In zone 4, operator  $G$  is constant. It represents the outside of the resulting object boundary, and since it has to be continuously (at least  $G^1$ ) joined with the other zones, an obvious

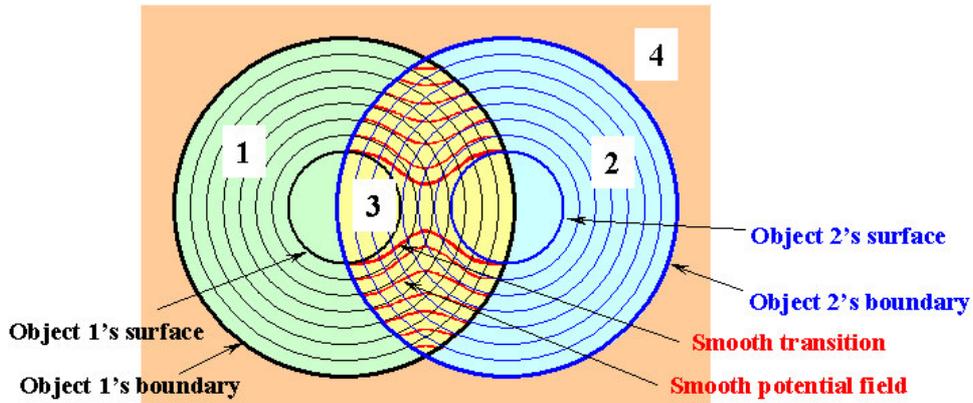


Fig. 3. Graph illustrating a 2D section of the potential fields when *two* spheres are combined (union) with smooth transition. Zone 1 is inside object 1 boundary and outside object 2 boundary, zone 2 is inside object 2 boundary and outside object 1 boundary, zone 3 is the intersection of objects' boundaries and zone 4 is outside both boundaries. Lines represent sections of iso-surfaces. In zone 3, we also show sections of the smooth transition.

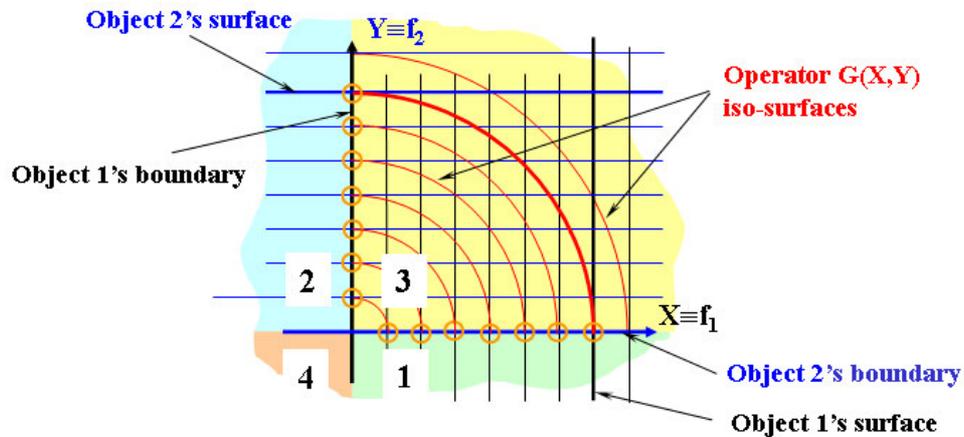


Fig. 4. Plot of the union binary composition operator  $G$  which generates the smooth composition shown in Figure 3. In order to better correspond to Figure 3, zone 1 can be reduced to the  $X$  axis ( $Y = 0, \forall X$ ), zone 2 can be reduced to the  $Y$  axis ( $X = 0, \forall Y$ ) and zone 4 can be reduced to the point  $(0, 0)$  ( $X = 0$  and  $Y = 0$ ).

value is:  $G(0, 0) = 0$ .

These fundamental statements give us a theoretical basis to provide Boolean composition operators for bounded implicit primitives. Note that the bounding box of the resulting object is easy to compute. For the union operator, the box is the union of those of the two combined primitives. For the intersection operator, it is their intersection and for the

difference it is the box of the object defined by function  $f_1$  (for the case  $object_1 \setminus object_2$ ).

We now emphasize the link between composition operators for unbounded primitives and bounded primitives. If we look at Figure 4 carefully, we see that the operator which combines bounded objects with smooth transition also combine the *zero*-isosurface with sharp transition (with a smooth potential field everywhere else). Hence, as done in<sup>30,17</sup>, it is desirable to study operators combining unbounded implicit primitives with sharp transition<sup>9,16,17</sup>. Note that those proposed in<sup>9</sup> as well as operators with smooth transition for unbounded objects do not satisfy the continuity conditions at the junction between the different zones.

#### 4. Generic arc-of-an-ellipse function

In this Section, we introduce the general form of an operator  $G$  which satisfies the conditions presented in the previous Section. It is based on a geometric construction of function  $G$  and a specific adaptation has already been used in<sup>16</sup> to combine unbounded implicit primitives with sharp transitions.

In order to respect the different constraints, operator  $G$  is piecewise defined. In zones 1 and 2, it returns the value of  $f_1$ , respectively  $f_2$  (as suggested in Section 3). In zone 4 it returns 0 and in zone 3, we propose to link the vertical iso-lines defined by  $G(f_1, 0) = f_1$  to the horizontal iso-lines defined by  $G(0, f_2) = f_2$  with a quarter of an ellipse (Figure 5) defined by the following equation:

$$\frac{(X_p - X_{p0})^2}{(C_p - X_{p0})^2} + \frac{(Y_p - Y_{p0})^2}{(C_p - Y_{p0})^2} = 1, \quad (4)$$

where a point  $P \in \mathbb{R}^2$  has the coordinates  $P(f_1 = X_p, f_2 = Y_p)$  and the potential in this point is  $G(P) = G(X_p, Y_p) = C_p$ . The center of the ellipse passing by the point  $P$  is the point  $P_0(X_{p0}, Y_{p0})$ .

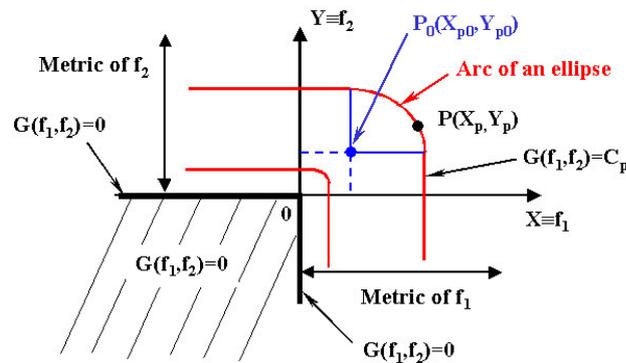


Fig. 5. Plot of the general form of a union composition operator  $G$ . The arc-of-an-ellipse links the horizontal and vertical half-lines with a  $G^1$  continuity.

8 *Loïc Barthe and Brian Wyvill and Erwin de Groot*

The junction between the constant map  $G(f_1, 0) = f_1$  (or  $G(0, f_2) = f_2$ ) and the quadratic arc-of-an-ellipse is  $G^1$  continuous and they are both internally  $G^\infty$ .

In Equation 4, unknowns  $X_{p0}$  and  $Y_{p0}$  are to be expressed in terms of  $C_p$  and the equation is solved to compute the value  $C_p$  returned by operator  $G$  at a point  $P(X_p, Y_p)$ . In the following Sections, we use *two* different geometric constructions based on this general definition in order to provide Boolean composition with and without smooth transition.

### 5. Operators with smooth transition

Our operator  $G$  is already designed to conserve the combined primitives' metrics outside the transition. All we have to do is to define boundaries for the arc-of-an-ellipse. For this purpose, we introduce *two* angles  $\theta_1$  and  $\theta_2$ , as illustrated in Figure 6, and it allows us to determine the unknowns  $X_{p0}$  and  $Y_{p0}$ :

$$X_{p0} = \frac{C_p}{\tan(\theta_2)} = C_p \cot(\theta_2) \text{ and } Y_{p0} = C_p \tan(\theta_1). \quad (5)$$

The adaptation of operator  $G$  to bounded primitives composition operators with smooth transitions is denoted  $\widetilde{G}_U$ .

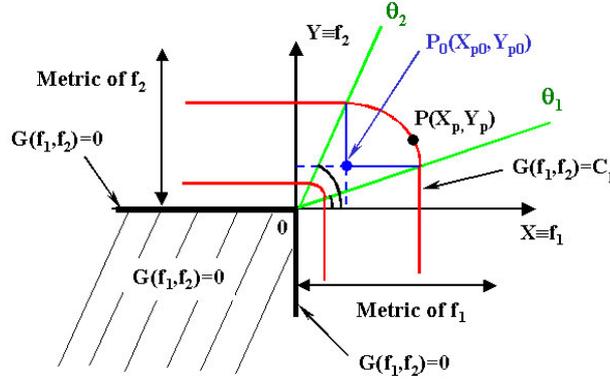


Fig. 6. Graph of our operator  $\widetilde{G}_U$ . Angles  $\theta_1$  and  $\theta_2$  are introduced in order to bound the arc-of-an-ellipse.

To allow accurate and intuitive control, the transition must be defined by control points on the  $C$  iso-potential surface. The first adaptation is to define angles  $\theta_1$  and  $\theta_2$  from the user Euclidean space  $\mathbb{R}^3$  by selecting control points  $p_1(x_1, y_1, z_1)$  and  $p_2(x_2, y_2, z_2)$  on the combined objects' surface, respectively  $f_1 = C$  and  $f_2 = C$  (Figure 7). We notice that points  $p_1$  and  $p_2$  must be selected inside the intersection of the objects' boundaries because, no transition can be defined outside these limits (as explained in Section 3). Our geometric construction of operator  $\widetilde{G}_U$  leads us to the following equation (equations of operators  $\widetilde{G}_\cap$  and  $\widetilde{G}_\setminus$  are given in Appendix A):

Points  $p_1 \in \mathbb{R}^3$  and  $p_2 \in \mathbb{R}^3$  selected by the user correspond to points  $P_1$  and  $P_2$  such as:  $P_1(C, f_2(p_1))$  and  $P_2(f_1(p_2), C)$ .

$$\theta_1 = \text{angle}([OX], [OP_1]), \quad \theta_2 = \text{angle}([OX], [OP_2])$$

$$\text{At point } P(X_p, Y_p) : \theta_p = \text{angle}([OX], [OP])$$

$$\widetilde{G}_U(X_p, Y_p) = \begin{cases} X_p & \text{if } Y_p = 0 \\ Y_p & \text{if } X_p = 0 \\ X_p & \text{if } \theta_p \leq \theta_1 \\ Y_p & \text{if } \theta_p \geq \theta_2 \\ C_p & \text{where } C_p \text{ is the solution of:} \\ & \frac{(X_p - C_p \cdot \cot(\theta_2))^2}{(C_p - C_p \cdot \cot(\theta_2))^2} + \frac{(Y_p - C_p \cdot \tan(\theta_1))^2}{(C_p - C_p \cdot \tan(\theta_1))^2} = 1 \\ & \text{if } \theta_p \in (\theta_1, \theta_2) \end{cases} \quad (6)$$

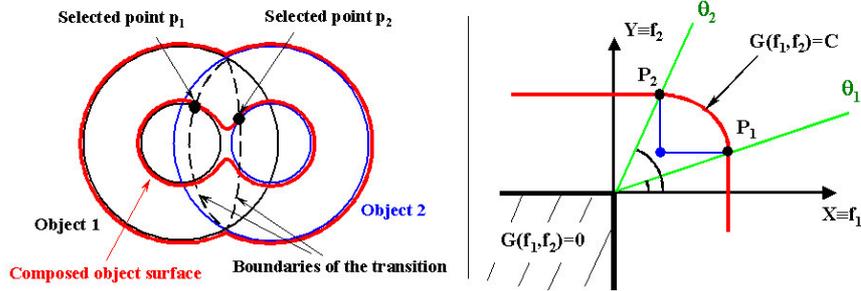


Fig. 7. Union with smooth transition controlled point-by-point in the user Euclidean space and its function representation.

The closed form solution for the evaluation of  $C_p$  in Equation 6 is given in <sup>16</sup>. With operator  $\widetilde{G}_U$  in this form, only the boundaries of the transition can be controlled. For fixed angles  $\theta_1$  and  $\theta_2$  it is necessary to be able to choose the smoothness of the transition. This leads us to add at least one control point. In fact, adding one or more control points brings us to the same solution. To conserve the  $G^1$  continuity in the field, we multiply  $\widetilde{G}_U(P)$  by a function  $m(\theta_p)$  where  $m$  is an interpolation function of  $\mathbb{R} \rightarrow \mathbb{R}$  when  $\theta_p \in [\theta_1, \theta_2]$  and  $m(\theta_p) = 1$  otherwise. A valid graph for function  $m$  is shown in Figure 8. The link with the control points is done as follows:  $m(\theta_1) = 1, m'(\theta_1) = 0$  and  $m(\theta_2) = 1, m'(\theta_2) = 0$  to ensure  $G^1$  continuity at the beginning and the end of the transition. Then  $k_i$  ( $i > 2$ ) are computed from the control points  $p_i(x_i, y_i, z_i)$  ( $i > 2$ ) selected in the Euclidean modelling space  $\mathbb{R}^3$ . Point  $p_i$  allows us to compute the point  $P_i(f_1(p_i), f_2(p_i))$ , followed by  $\theta_{P_i}$  and

$C_i = \widetilde{G}_U(P_i)$ . The corresponding point  $k_i$  ( $i > 2$ ), to interpolate, has then the coordinates:  $k_i(\theta_i, C/C_i)$ . We have chosen one-dimensional cubic polynomial splines<sup>31</sup> to define function  $m$  when  $\theta_p \in [\theta_1, \theta_2]$  for their adequate smoothness and oscillation properties, and for their inexpensive computation cost. We finally obtain the union Boolean operator with “functionally defined” transitions for bounded objects in Equation 7.

$$\widetilde{G}_U^{final}(P) = m(\theta_p) \cdot \widetilde{G}_U(P) \quad (7)$$

The same path has been followed to build the intersection Boolean operator with “functionally defined” transition  $\widetilde{G}_\cap^{final}(P) = m(\theta_p) \cdot \widetilde{G}_\cap(P)$  from  $\widetilde{G}_\cap(P)$ .

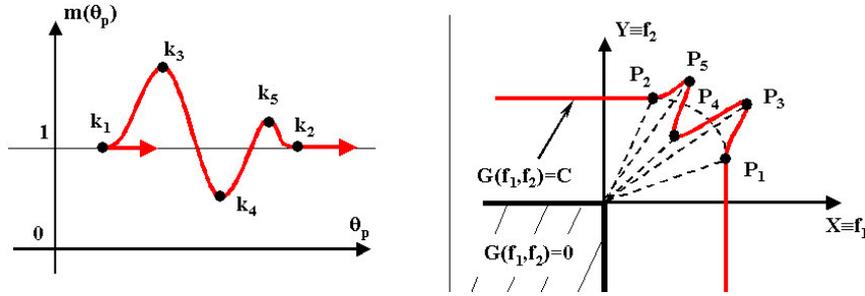
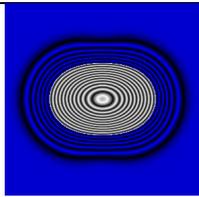
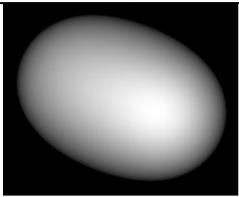
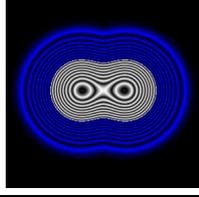
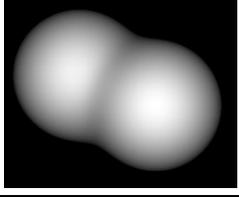
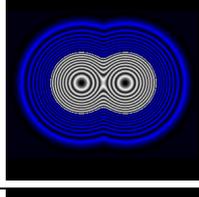
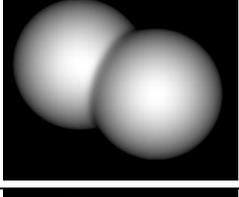
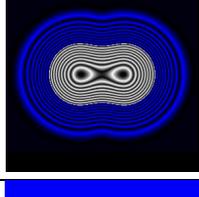
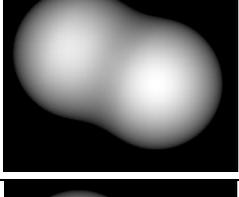
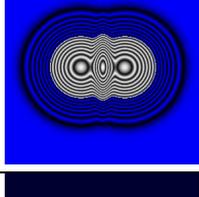
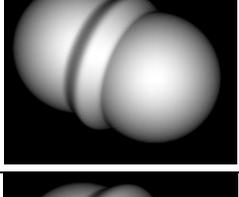
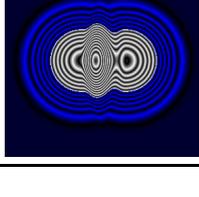
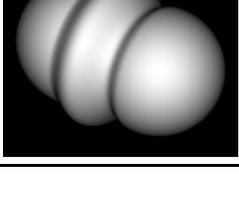


Fig. 8. Graph of an interpolation function  $m(\theta_p)$  used to deform the operator  $\widetilde{G}_U$  and allow the control point-by-point of the transition.

The difference operator  $\widetilde{G}_\setminus$  cannot be directly derived with the standard method used for unbounded primitives:  $\widetilde{G}_\setminus(f_1, f_2) = \widetilde{G}_\cap(f_1, -f_2)$ . Ricci<sup>6</sup> proposed the realization of the difference operator on bounded implicit objects using the intersection operator applied on  $f_1$  and  $(2C - f_2)$  instead of  $f_2$ . The same method used on our intersection operator  $\widetilde{G}_\cap^{final}$  gives the difference Boolean operator with “functionally defined” transition:  $\widetilde{G}_\setminus^{final}$ . Figure 9(c) shows a ring object built from the ring of Figure 9(a) and a gem similar to that in Figure 9(b). The gem has been further modified by two profile curves. The first profile curve modifies the smooth intersection operation used to construct the gem. The second modifies the difference operation between the gem and a sphere (implicit point primitive) at its center. Finally, another sphere has been added with a smooth union operation. The gem is then joined to the ring using another smooth union with a profile curve. Table 1 illustrates different union composition operators with smooth transition and allows us to compare of the computation times, the potential field variations and the shape produced at the transition level. The increase of the evaluation cost of our operators is compensated by the controllability of the form of the transition.

A function  $m$  of  $\mathbb{R} \rightarrow \mathbb{R}$  is used to provide point-by-point control. Because such a function must be single valued, we do not obtain a true free-form control of the transition (as proposed in<sup>16</sup> for unbounded primitives). However, we deal with bounded objects. The transition in operators like ours is then essentially used to smooth the junction of

Table 1. Time in milliseconds to compute potential function values for a  $128^3$  grid (2097152 evaluations), using a 1.0 GHz AMD Athlon processor with 512 Mbytes of DDR memory. (a) Using Ricci's operator with  $n = 1$ , (b) Ricci's operator with  $n = 3$ , (c) Ricci's operator with  $n = 7$ , (d) operator  $\widetilde{g}_{\cup}$ , (e) operator  $\widetilde{g}_{\cup}^{final}$  with 3 control points and (f) operator  $\widetilde{g}_{\cup}^{final}$  with 5 control points. The middle column shows two-dimensional sections of the full grid.

time	2D section	3D object
(a) 246		
(b) 884		
(c) 940		
(d) 1005		
(e) 1933		
(f) 2315		

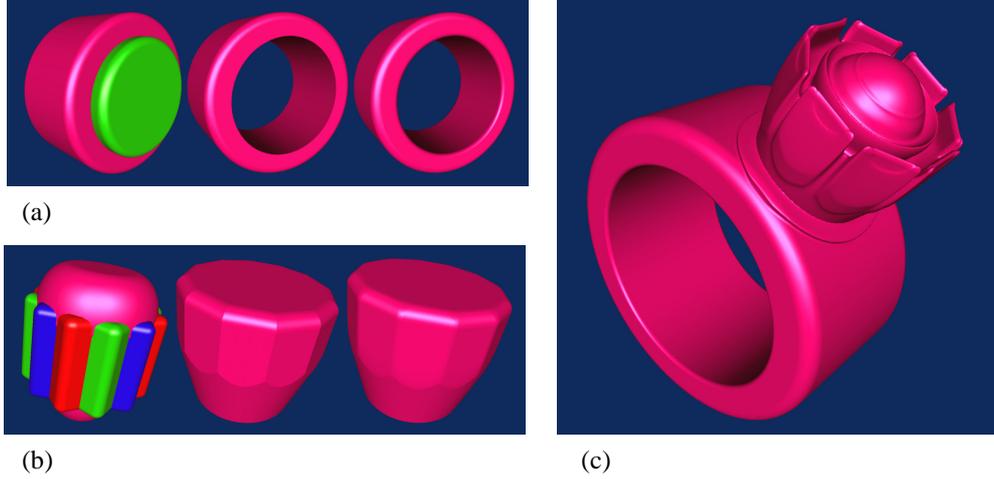
12 *Loïc Barthe and Brian Wyvill and Erwin de Groot*

Fig. 9. (a) The ring is built using two implicit cylinders and applying subtraction, the center image uses the Ricci CSG subtraction operator, the right hand image, our smooth CSG subtraction  $\widehat{G}_\setminus$ . (b) The gemstones are built from implicit *box* primitives and an implicit cone, center image uses the Ricci intersection operator, right hand image our smooth intersection  $\widehat{G}_\cap$ . (c) Applying the smooth CSG operators  $\widehat{G}_\cup$ ,  $\widehat{G}_\cap$ ,  $\widehat{G}_\setminus$  and profile curve operators  $\widetilde{G}_\cap^{final}$ ,  $\widetilde{G}_\setminus^{final}$  on bounded implicit objects.

two primitives when they are combined. We assume that in a general case, to create the desired smooth transition, three, four or five control points give enough flexibility. Free-form curves are needed in very specific cases, and often it remains easier to build a new primitive and to combine it with a smooth transition.

## 6. Operators with sharp transition

Our operator  $G$  combines the primitives' boundaries with sharp transition and generates a smooth  $G^1$  potential field everywhere else using an arc-of-an-ellipse. Since we want to combine the object surfaces with sharp transition, we have to provide a solution where operator  $G$  does not smooth both boundaries and  $C$  iso-surfaces, but still smooth the potential field everywhere else. In order to satisfy this additional constraint, we propose the geometric construction illustrated in Figure 10 and we denote our binary union operator with sharp transition as  $\widehat{G}_\cup$ . Note that we cannot derive a simple solution using the operators proposed in<sup>30,17</sup> because they generate discontinuities in the field around the origin  $(0, 0)$ , at the level of the X and Y axes (see Figure 4).

This geometric representation leads us to the following definition of unknowns  $X_{p0}$  and  $Y_{p0}$ :

$$\text{In zone } a: X_{p0} = Y_{p0} = \frac{C_p^2}{C} = C_a C_p^2 \text{ with } C_a = \frac{1}{C} \quad (8)$$

$$\text{in zone } b: X_{p0} = Y_{p0} = \sqrt{CC_p}. \quad (9)$$

Operators  $\widehat{G}_\cap$  and  $\widehat{G}_\setminus$  are presented in Appendix B and the closed form solution for the

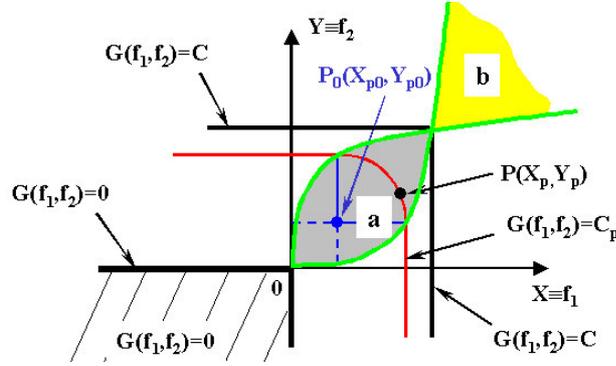


Fig. 10. Graph of our operator  $\widehat{G}_U$ . Both boundaries and  $C$  iso-surfaces are combined with sharp transition while the potential field is  $G^1$  continuous everywhere else.

evaluation of  $C_p$  in Equation 10 is given in Appendix C. Figure 11 illustrates the effect of operator  $\widehat{G}_U$  on both shape and potential field, and Figure 2 shows its smoothing property when the resulting object is to be combined with smooth transition.

$$\widehat{G}_U(X_p, Y_p) = \begin{cases} X_p & \text{if } Y_p = 0 \\ Y_p & \text{if } X_p = 0 \\ Y_p & \text{if } Y_p \geq \sqrt{CX_p} \text{ and } Y_p \leq C \\ X_p & \text{if } Y_p \leq CaX_p^2 \text{ and } X_p \leq C \\ Y_p & \text{if } Y_p \geq CaX_p^2 \text{ and } Y_p > C \\ X_p & \text{if } Y_p \leq \sqrt{CX_p} \text{ and } X_p > C \\ C_p & \text{where } C_p \text{ is the solution of:} \\ & \frac{(X_p - CaC_p^2)^2 + (Y_p - CaC_p^2)^2}{(C_p - CaC_p^2)^2} = 1 \\ & \text{if } P(X_p, Y_p) \text{ is in zone } a \\ C_p & \text{where } C_p \text{ is the solution of:} \\ & \frac{(X_p - \sqrt{CC_p})^2 + (Y_p - \sqrt{CC_p})^2}{(C_p - \sqrt{CC_p})^2} = 1 \\ & \text{if } P(X_p, Y_p) \text{ is in zone } b \end{cases} \quad (10)$$

The evaluations of  $\widehat{G}_U$  for a  $128^3$  grid (2097152 evaluations), using a 1.0 GHz AMD Athlon processor with 512 Mbytes of DDR memory takes 2010 milliseconds (to create the object of the central image in Figure 11). We notice that we first store the combined primitives in regular grids. Therefore, the given time does not depend on the complexity of the combined objects, and the composition of *two* complex primitives leads us to the same result. What is time consuming is the complexity and the expensive computation of  $\widehat{G}_U$  when it has to be evaluated in zones *a* and *b*. However, we are dealing with bounded objects, and they do not need to be evaluated at a point of  $\mathbb{R}^3$  which is located outside their bounding-box. Moreover, if the point is inside the box, the expensive evaluation occurs

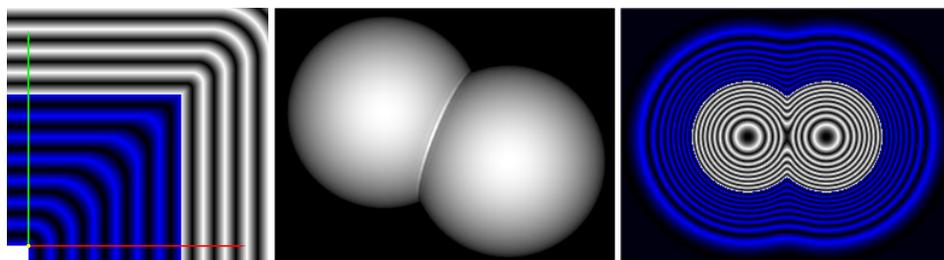


Fig. 11. From left to right: Plot of our operator  $\widehat{G}_U$ , its effect on the composition of *two* implicit bounded spheres and a 2D section of the generated potential field. In the left and right Figures, the white area represents the inside of the object and the dark area, the outside.

only if the point is also inside the intersection of the combined objects boundaries. This significantly reduces the complexity brought by our operator when it is used to model a complex object.

## 7. Example

Figure 12 shows a teddy bear made of some simple primitives and the binary Boolean composition operators described in this paper. Figure 13 shows an arm of the teddy bear. The arm is the smooth difference between a scaled point primitive and a plane primitive. The shape of the end of the arm is created by placing controlpoints. Figure 14 shows an ear of the teddy bear. The ear is the sharp difference between a point primitive and the sharp union of a sphere and a plane. Once created, the ear is combined to the head with a smooth transition.

We point out that it is obvious that our operators are computationally expensive (as shown in Table 1). However, they have the fundamental property to be bounded, and hence, our operators have to be evaluated only in potential field areas where implicit primitives are combined. In other part of the space, the computational complexity remains the one of the combined primitives themselves.

Figures 9, 12, 13 and 14 are computed using a standard ray marching algorithm without any optimisation. The computation of Figure 12 took ten hours on an Intel PIII 1.3Ghz. The use of a faster raytracing technique will significantly reduce this time, but fast rendering is out of the scope of this paper.

## 8. Conclusion

We have presented a theoretical background which gives us a basis for the construction of binary Boolean composition operators for bounded implicit primitives. Boolean composition operators with smooth or sharp transition can be derived while preserving a  $G^1$  continuous potential field. The operators  $\widetilde{G}^{final}$  greatly increase accuracy and freedom in the control of the transition when primitives are smoothly combined. The operators  $\widehat{G}$  limit the discontinuities in the composed objects to a minimum, so these objects can be used in



Fig. 12. Teddy bear made from simple primitives and the operations described in this paper

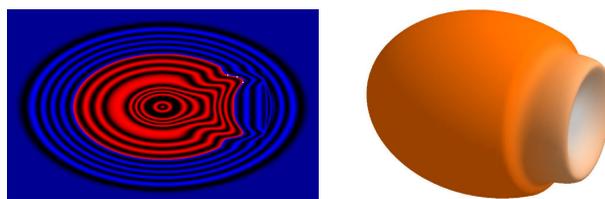


Fig. 13. An arm of the teddy bear. White circles indicate controlpoints.



Fig. 14. An ear of the teddy bear.

other operations without generating  $G^1$  discontinuities in the transition area, so that even after several Boolean compositions, the resulting object still have the “automatic blending” property.

However, our operators remain expensive to evaluate and they are still limited to binary compositions. Hence, we can say that these results provide a basis to study new composition operators for bounded implicit primitives, and improvements can be investigated in order to propose fast evaluated operators or operators satisfying specific properties like the Lipschitz<sup>32</sup> condition to accelerate the rendering for example. The extension from binary operators to n-ary operators is also interesting. However, we did not find any direct derivation of our operators in order to define n-ary operators and the definition of n-ary operators providing both accurate control and smooth transition seems to be a challenging problem.

### Acknowledgements

This work has been partially funded by E.U. through the MINGLE project. Contract HPRN-CT-1999-00117

### Appendix A. Operators $\widetilde{G}_\cap$ and $\widetilde{G}_\setminus$

- Operator  $\widetilde{G}_\cap$ :

Points  $p_1 \in \mathbb{R}^3$  and  $p_2 \in \mathbb{R}^3$  selected by the user correspond to points  $P_1$  and  $P_2$  such as:  $P_1(f_2(p_1), C)$  and  $P_2(C, f_1(p_2))$ .

$$\theta_1 = \text{angle}([OX], [OP_1]), \quad \theta_2 = \text{angle}([OX], [OP_2])$$

$$\text{At point } P(X_p, Y_p) : \theta_p = \text{angle}([OX], [OP])$$

$$\widetilde{G}_\cap(X_p, Y_p) = \begin{cases} 0 & \text{if } Y_p = 0 \\ 0 & \text{if } X_p = 0 \\ Y_p & \text{if } \theta_p \leq \theta_1 \\ X_p & \text{if } \theta_p \geq \theta_2 \\ C_p & \text{where } C_p \text{ is the solution of:} \\ & \frac{(X_p - C_p \cdot \cot(\theta_1))^2}{(C_p - C_p \cdot \cot(\theta_1))^2} + \frac{(Y_p - C_p \cdot \tan(\theta_2))^2}{(C_p - C_p \cdot \tan(\theta_2))^2} = 1 \\ & \text{if } \theta_p \in ]\theta_1, \theta_2[ \end{cases} \quad (\text{A.1})$$

- Operator  $\widetilde{G}_\setminus$ :

Operator  $\widetilde{G}_\setminus$  is directly obtained from operator  $\widetilde{G}_\cap$  using the following expression:

$$\widetilde{G}_\setminus(f_1, f_2) = \widetilde{G}_\cap(f_1, 2C - f_2) \quad (\text{A.2})$$

## Appendix B. Operators $\widehat{G}_\cap$ and $\widehat{G}_\setminus$

- Operator  $\widehat{G}_\cap$ :

$$\widehat{G}_\cap(X_p, Y_p) = \begin{cases} 0 & \text{if } Y_p = 0 \\ 0 & \text{if } X_p = 0 \\ X_p & \text{if } Y_p \geq \sqrt{CX_p} \text{ and } X_p \leq C \\ Y_p & \text{if } Y_p \leq C_a X_p^2 \text{ and } Y_p \leq C \\ X_p & \text{if } Y_p \geq C_a X_p^2 \text{ and } X_p > C \\ Y_p & \text{if } Y_p \leq \sqrt{CX_p} \text{ and } Y_p > C \\ C_p & \text{where } C_p \text{ is the solution of:} \\ & \frac{(X_p - \sqrt{CC_p})^2 + (Y_p - \sqrt{CC_p})^2}{(C_p - \sqrt{CC_p})^2} = 1 \\ & \text{if } P(X_p, Y_p) \text{ is in zone } a \\ C_p & \text{where } C_p \text{ is the solution of:} \\ & \frac{(X_p - C_a C_p^2)^2 + (Y_p - C_a C_p^2)^2}{(C_p - C_a C_p^2)^2} = 1 \\ & \text{if } P(X_p, Y_p) \text{ is in zone } b \end{cases} \quad (\text{B.1})$$

- Operator  $\widehat{G}_\setminus$ :

Operator  $\widehat{G}_\setminus$  is directly obtained from operator  $\widehat{G}_\cap$  using the following expression:

$$\widehat{G}_\setminus(f_1, f_2) = \widehat{G}_\cap(f_1, 2C - f_2) \quad (\text{B.2})$$

## Appendix C. Closed form solution for the evaluation of $C_p$ in our new composition operators with sharp transition

**Solution for the equation:**

$$\frac{(X_p - C_a C_p^2)^2 + (Y_p - C_a C_p^2)^2}{(C_p - C_a C_p^2)^2} = 1 \quad (\text{C.1})$$

$C_p$  is one of the roots of the following equation:

$$C_a^2 C_p^4 + 2C_a C_p^3 - (2C_a X_p + 2C_a Y_p + 1)C_p^2 + X_p^2 + Y_p^2 = 0 \quad (\text{C.2})$$

$C_p$  is computed with:

$$\begin{aligned} S_{11} &= 48C_a^3 Y_p^2 X_p + 64C_a^3 Y_p^3 + 78C_a^2 Y_p^2 + 64C_a^3 X_p^3 + 48C_a^3 X_p^2 Y_p + 78C_a^2 X_p^2 - 24C_a^2 X_p Y_p - 6C_a X_p - 6C_a Y_p - 1 \\ S_{12} &= -3(X_p^2 + Y_p^2)(-64C_a^3 X_p^3 + 16C_a^3 X_p^2 Y_p - 35C_a^2 X_p^2 + 64C_a^4 X_p^2 Y_p^2 + 72C_a^2 X_p Y_p) \\ S_{13} &= -3(X_p^2 + Y_p^2)(16C_a^3 Y_p^2 X_p + 14C_a X_p + 14C_a Y_p - 64C_a^3 Y_p^3 - 35C_a^2 Y_p^2 + 2) \\ S_1 &= S_{11} + 6C_a \sqrt{S_{12} + S_{13}} \\ S_2 &= \frac{5\sqrt[3]{S_1} + 4\sqrt[3]{S_1} C_a X_p + 4\sqrt[3]{S_1} C_a Y_p + \sqrt[3]{S_1^2} + 16C_a^2 Y_p^2 + 16C_a^2 X_p^2 + 8C_a^2 X_p Y_p + 4C_a X_p + 4C_a Y_p + 1}{\sqrt[3]{S_1}} \\ S_3 &= \sqrt{S_2} \sqrt[3]{S_1} C_a Y_p \\ S_4 &= \sqrt{S_2} \sqrt[3]{S_1} C_a X_p \end{aligned}$$

18 *Loïc Barthe and Brian Wyvill and Erwin de Groot*

$$\begin{aligned}
S_5 &= \sqrt{S_2} \sqrt[3]{S_1} \\
S_6 &= \sqrt{S_2} \sqrt[3]{S_1^2} \\
S_7 &= \sqrt[3]{S_1} \\
R_1 &= -10S_5 - 8S_4 - 8S_3 + S_6 + 16\sqrt{S_2}C_a^2Y_p^2 + 16\sqrt{S_2}C_a^2X_p^2 + 8\sqrt{S_2}C_a^2X_pY_p + 4\sqrt{S_2}C_aX_p + 4\sqrt{S_2}C_aY_p + \sqrt{S_2} \\
R_2 &= 12S_7(C_aX_p + C_aY_p + 1) \\
C_p &= \frac{1}{6C_a} \left( -3 + \sqrt{3S_2} - \sqrt{\frac{-3(R_1 + R_2)}{S_5}} \right) \tag{C.3}
\end{aligned}$$

We notice that in the previous equation, values  $S_i$  and  $R_1$  can be complex. Therefore, all the computations have to be done with complex numbers, even if the result  $C_p$  is real.

**Solution for the equation:**

$$\frac{(X_p - \sqrt{CC_p})^2 + (Y_p - \sqrt{CC_p})^2}{(C_p - \sqrt{CC_p})^2} = 1 \tag{C.4}$$

$C_p$  is one of the roots of the following equation:

$$-C_p^2 + 2C_p\sqrt{CC_p} + CC_p - 2(X_p + Y_p)\sqrt{CC_p} + X_p^2 + Y_p^2 = 0 \tag{C.5}$$

$C_p$  is computed with:

$$\begin{aligned}
S_{11} &= -18C^5X_p - 18C^5Y_p - 36C^4X_p^2 - 36C^4Y_p^2 + 108C^4X_pY_p - C^6 \\
S_{12} &= 18C^8X_p^3Y_p - 72C^7Y_p^3X_p^2 - 36C^7X_pY_p^4 - 3C^{10}X_pY_p + 18C^8X_pY_p^3 - 48C^9X_pY_p^2 \\
S_{13} &= 36C^6X_p^2Y_p^4 - 36C^7X_p^5 + 3C^9X_p^3 + 42C^8X_p^4 + 3C^9Y_p^3 + 42C^8Y_p^4 - 36C^7Y_p^5 \\
S_{14} &= 12C^6X_p^6 + 12C^6Y_p^6 - 72C^7X_p^3Y_p^2 - 48C^9X_p^2Y_p + 165C^8X_p^2Y_p^2 + 36C^6X_p^4Y_p^2 - 36C^7Y_pX_p^4 \\
S_1 &= S_{11} + 12\sqrt{S_{12} + S_{13} + S_{14}} \\
S_3 &= \sqrt[3]{S_1} \\
S_2 &= (15C^2S_3 + 3S_3^2 + 36C^3X_p + 36C^3Y_p - 36C^2X_p^2 - 36C^2Y_p^2 + 3C^4)/S_3 \\
S_4 &= \sqrt{S_2} \\
R_1 &= 10C^2S_3S_4 - S_4S_3^2 - 12C^3X_pS_4 - 12C^3Y_pS_4 + 12C^2X_p^2S_4 + 12C^2Y_p^2S_4 - C^4S_4 \\
R_2 &= 36C^3S_3 - 36C^2X_pS_3 - 36C^2Y_pS_3 \\
C_p &= \frac{1}{C} \left( \frac{C}{2} - \frac{S_4}{6} + \frac{1}{6} \sqrt{\frac{3(R_1 - R_2)}{S_3S_4}} \right)^2 \tag{C.6}
\end{aligned}$$

We notice that in the previous equation, some values can be complex. Therefore, all the computations have to be done with complex numbers, even if the result  $C_p$  is real.

## References

1. J. Bloomenthal (ed), *Introduction to Implicit Surfaces*, (Morgan-Kaufmann, 1997).
2. M. Chen, A. Kaufman and R. Yagel (eds), *Volume Graphics*, (Springer, London, 2000).

3. J.F. Blinn, "A Generalization of algebraic surface drawing", *ACM Transaction on Graphics*, **1**(3), (1982) pp. 235–256.
4. J. Bloomenthal and B. Wyvill, "Interactive techniques for Implicit Modeling", *Computer Graphics (Proc. of SIGGRAPH 1990)*, **24**(2), (1990) pp. 109–116.
5. A. P. Rockwood, "The Displacement Method for Implicit Blending Surfaces in Solid Models", *ACM Transaction on Graphics*, **8** (4), (1989) pp. 279–297.
6. A. Ricci, "A Constructive Geometry for Computer Graphics", *The Computer Journal*, **16**(2) (1973) pp. 157–160.
7. M.P. Gascuel, "An Implicit Formulation for Precise Contact Modeling Between Flexible Solids", *Computer Graphics (Proc. of SIGGRAPH 1993)*, (1993) pp. 313–320.
8. V. Shapiro, "Real functions for representation of rigid solids", *Computer Aided Geometric Design*, **11**(2) (1994) pp. 153–175.
9. A. Pasko, V. Adzhiev, A. Sourin and V. Savchenko, "Function Representation in Geometric Modeling: Concepts, Implementation and Applications", *The Visual Computer*, **8**(2) (1995) pp. 429–446.
10. B. Wyvill and K.v. Overveld, "Polygonization of Implicit Surfaces with Constructive Solid Geometry", *Journal of Shape Modelling*, **2**(4) (1996) pp. 257–274
11. H. Nishimura, M. Hirai, T. Kawai, I. Shirakara and K. Omura, "Object Modeling by Distribution Functions", *Electronics Communications (in Japanese)*, **J68**(D) (1985) pp. 718–725.
12. G. Wyvill, C. McPheeters and B. Wyvill, "Data Structure for Soft Objects", *The Visual Computer*, **2**(4) (1986) pp. 227–234.
13. V. Adzhiev, M. Kazakov and A. Pasko, "Hybrid System Architecture for Volume Modeling", *Computer and Graphics*, **24**(1) (2000) pp. 67–78.
14. A. Sourin and A. Pasko, "Function Representation for Sweeping by a Moving Solid", *IEEE Transaction on Visualization and Computer Graphics*, **2**(1) (1996) pp. 11–18.
15. G. Pasko, A. Pasko, M Ikeda and T. Kunii, "Bounded Blending Operations", *Proc. of Shape Modeling International 2002*, (2002) pp. 95–103.
16. L. Barthe, N.A. Dodgson, M.A. Sabin, B. Wyvill and V. Gaildrat, "Two-Dimensional Potential Fields for Advanced Implicit Modeling Operators", *Computer Graphics Forum*, **22**(1) (2003) pp. 23–33.
17. P.C. Hsu and C. Lee, "The Scale Method for Blending Operations in Functionally-Based Constructive Geometry", *Computer Graphics Forum*, **22**(2) (2003) pp. 143–158.
18. M. Chen and J. Tucker, "Constructive Volume Geometry", *Computer Graphics Forum*, **19**(4) (2000) pp. 281–293.
19. L. Barthe, B. Mora, N.A. Dodgson and M.A. Sabin, "Interactive Implicit Modelling Based on  $C^1$  Continuous Reconstruction of Regular Grids", *International Journal of Shape Modeling*, **8**(2) (2002) pp. 99–117.
20. Z. Kacic-Alesic and B. Wyvill, "Controlled Blending of Procedural Implicit Surfaces", *Proc. of Graphic Interface 1991*, (1991) pp. 236–245.
21. C. Blanc and C. Schlick, "Extended Field Functions for Soft Objects", *Proc. of Implicit Surfaces 1995*, (1995) pp. 21–32.
22. P.C. Hsu and C. Lee, "Field Functions for Blending Range on Soft Objects", *Computer Graphics Forum*, Proc. of EUROGRAPHICS 2003, **22**(3) (2003).
23. B. Wyvill, A. Guy and E. Galin, "Extending the CSG Tree: Warping, Blending and Boolean Operations in an Implicit Surface Modeling System", *Computer Graphics Forum*, **18**(2) (1999) pp. 149–158.
24. E. Galin and S. Akkouche, "Incremental Polygonization of Implicit Surfaces", *Graphical Models*, **62**(1) (2000) pp. 19–39.
25. L. Barthe, V. Gaildrat and R. Caubet, "Extrusion of 1D implicit profiles: Theory and first application", *International Journal of Shape Modeling*, **7**(2) (2001) pp. 179–199.

20 *Loïc Barthe and Brian Wyvill and Erwin de Groot*

26. B. Crespin, C. Blanc and C. Schlick, "Implicit Sweep Objects", *EUROGRAPHICS 1996*, **15**(3) (1996) pp. 165–174.
27. M. Desbrun and M.P. Gascuel, "Animating Soft Substances with Implicit Surfaces", *Computer Graphics (Proc. of SIGGRAPH 1995)*, (1995) pp. 287–290.
28. A. Guy and B. Wyvill, "Controlled Blending for Implicit Surfaces Using a Graph", *Proc. of Implicit Surfaces 1995*, (1995) pp. 107–112.
29. E. Ferley, M. P. Cani and J. D. Gascuel, "Practical Volumetric Sculpting", *The Visual Computer*, **16**(8) (2000) pp. 469–480.
30. L. Barthe, N.A. Dodgson, M.A. Sabin, B. Wyvill and V. Gaildrat, "Different applications of two-dimensional potential fields for volume modeling", *technical report UCAM-CL-TR-541, ISSN 1476-2986*, University of Cambridge, UK, (2002).
31. I.D. Faux and M.J. Pratt, "Computational Geometry for Design and Manufacture", *Ellis Horwood*, (1979).
32. D. Kalra and A.H. Barr, "Guaranteed Ray Intersections with Implicit Surfaces", *SIGGRAPH '89 Proceedings*, **23**(3) (1989) pp. 297–306.