

On Improving the Search for Critical Points of Implicit Functions

Wu, Shin – Ting*

Marcelo de Gomensoro Malheiros

Image Computing Group (GCI/DCA/FEEC)
State University of Campinas, São Paulo, Brazil

Abstract

Recently, there is a trend to develop efficient polygonization techniques for implicit surfaces, aiming at interactive modeling and animation. One of the most challenging issues for such techniques is to ensure the matching between the topology of the surface and the topology of its polygonal approximation. Therefore, it is important to accurately extract the topological information from a given implicit function. This can be done using Morse theory, which says that the critical points of a real function are intimately related to the topology of its level sets. Previous efforts to locate critical points consist of either employing interval search over the domain of the implicit function or trying to find them incrementally in the neighborhood of selected vertices. In this paper we propose an alternative procedure for a specific skeleton-based model. We use the spatial coherence of the skeleton elements to give a good first guess for the locations of the critical points. Newton’s method is then applied to improve the accuracy of these predictions.

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

Keywords: critical points, implicit surfaces, Morse theory, topology.

1 Introduction

Implicit surfaces have been widely used in the last few years for modeling, visualization, and animation. Of particular interest is their inherent capability of representing smooth shapes with arbitrary topology. Moreover, implicit formulations are well suited to ray-tracing algorithms.

We are focused on a specific modeling technique, where a surface is defined implicitly by a skeleton. Informally, a skeleton is a set of primitive functions which are combined to define the shape of an object. The skeleton-based approach provides a very flexible way to model surfaces, whose geometry can be altered by simply changing either the position or the shape of their building primitives.

However, implicit surfaces have a drawback: they cannot be easily rendered at interactive rates by ray-tracing methods. This

poses a critical limitation to real time modeling and animation systems. Consequently, much effort was put into creating approximative models, either using punctual sampling or polygonal representations.

Techniques that only provide sample points on the implicit surface are adequate for simple objects. On the other hand, for complex scenes with several overlapping objects, meaningful visualization becomes more difficult. In this case, opaque visualization methods employing polygonal approximations have shown to be more convenient [18].

One of the most promising polygonization techniques was presented by Desbrun et al. [6, 7] and later refined by Crespín et al. [5]. It uses a loosely constrained particle system to create a polygonal patch in the neighborhood of each primitive that belongs to the skeleton. Those patches are then “zippered” together to create a global mesh that approximates the corresponding implicit surface.

This divide-and-conquer approach has many benefits over other polygonization techniques, namely its efficiency and local control over sampling. However, the topological correctness of the final mesh can only be guaranteed with additional knowledge about the topology of the implicit surface. By using Morse theory, we can obtain this knowledge by finding and classifying the critical points of the implicit function.

In this paper we address the issue of determining the critical points of a particular class of implicit functions. Instead of using a generic interval search [11, 18] or monitoring the rate of change of the implicit function [4], we use the spatial coherence of the skeleton elements to predict the sites of the critical points. The obtained estimates are then improved by using Newton’s method. We show that these predictions are tight in the sense that they normally converge with few iterations.

The remainder of this paper is organized as follows. We begin with related work and some basic concepts. Then, in Section 3, we describe the implicit model we worked with. In the next section we present our procedure to locate critical points and explain the reasoning behind it. Some representative examples are shown in Section 5 to illustrate the search process. Finally, concluding remarks are given.

2 Related Work

Given a function $F : \mathbb{R}^3 \rightarrow \mathbb{R}$ and a real value c , the set of points $\mathcal{S}_c$ that satisfy $F(\mathbf{x}) = c$ is called an *implicit surface*. The value c is named *threshold*. In fact, this surface corresponds to the level set of F associated with c , that is, $\mathcal{S}_c = F^{-1}(c)$.

Although the function F can be arbitrary, it is typically built on the basis of a *skeleton* [1, 21]. A skeleton is a set of simple pre-defined geometrical elements. To each such element is associated an *implicit primitive* f_i , which defines a scalar field in $\mathbb{R}^3$. Because each primitive can be evaluated and modified independently, we have control of the shape of the surface in the neighborhood of each primitive. Hence, the skeleton-based approach provides a concise yet flexible framework for designing implicit surfaces.

We can distinguish two basic techniques for interactive visualization of implicit surfaces: punctual sampling and polygonization.

*Address: DCA/FEEC/UNICAMP, Postal Box 6101, Campinas, SP, Brazil, ting@dca.fee.unicamp.br

In [2], Bloomenthal and Wyvill suggested that particle systems could be used to sample implicit surfaces, providing simple visualization with clouds of points. Later, Figueiredo et al. [8, 9] improved the idea using physically-based methods. They employed a modified gradient of F to define a vector field in $\mathbb{R}^3$, so that randomly scattered particles could move towards a specific level set. To achieve a more uniform sampling distribution, a mutual repulsion force among particles could be added to the simulation [19]. Witkin and Heckbert [20] and Desbrun et al. [6] used, respectively, discs and scales instead of points to enhance the 3D perception of the samples. Malheiros and Wu [12] got similar results using fog.

Despite those efforts, it is recognized that a set of sample points is not sufficient for correctly conveying the shape of objects in a complex scene [18]. Polygonal approximations are still more adequate for the visualization of elaborate models, which can take advantage of hidden surface removal techniques.

Several techniques for the polygonization of implicit surfaces have been proposed [3], but only few of them ensure the topological correctness of the resulting mesh. Of particular interest for us is the work of Bottino et al. [4] and Hart et al. [11, 18]. The foundation of their work is provided by Morse theory.

Morse theory was developed in the early sixties by Milnor [14]. It establishes the relationship between the critical points of F on a manifold $\mathcal{M}$ and the homotopy type of $\mathcal{M}$. Moreover, it states that the topological changes on the level sets of F occur only at critical points [10].

A *critical point* of a real function F is a point $\hat{\mathbf{x}}$ whose gradient vanishes, that is,

$$\nabla F(\hat{\mathbf{x}}) = (F_x(\hat{\mathbf{x}}), F_y(\hat{\mathbf{x}}), F_z(\hat{\mathbf{x}})) = \mathbf{0}.$$

The value $F(\hat{\mathbf{x}})$ of the function at a critical point is called a *critical value*.

The *Hessian* of F is defined as the Jacobian of its gradient,

$$H(\mathbf{x}) = J(\nabla F(\mathbf{x})) = \begin{bmatrix} F_{xx}(\mathbf{x}) & F_{xy}(\mathbf{x}) & F_{xz}(\mathbf{x}) \\ F_{yx}(\mathbf{x}) & F_{yy}(\mathbf{x}) & F_{yz}(\mathbf{x}) \\ F_{zx}(\mathbf{x}) & F_{zy}(\mathbf{x}) & F_{zz}(\mathbf{x}) \end{bmatrix}.$$

A value of λ , for which $(H(\mathbf{x}) - \lambda I)\mathbf{y} = \mathbf{0}$ has a solution $\mathbf{y} \neq \mathbf{0}$, is called an *eigenvalue* of $H(\mathbf{x})$. The corresponding solutions $\mathbf{y} \neq \mathbf{0}$ are called *eigenvectors* of $H(\mathbf{x})$.

Each critical point can be classified by analyzing the eigenvalues of its Hessian. Given a critical point $\hat{\mathbf{x}}$, we can obtain the three eigenvalues $l_1 \leq l_2 \leq l_3$ of $H(\hat{\mathbf{x}})$. If at least one of them is zero, the point is said to be *degenerate*. Otherwise, it is called *non-degenerate*, and we may characterize $\hat{\mathbf{x}}$ by the signs of its eigenvalues:

l_1	l_2	l_3	classification
-	-	-	maximum
-	-	+	2-saddle
-	+	+	1-saddle
+	+	+	minimum

Let us illustrate this concept with a height field defined by a scalar function over $\mathbb{R}^2$, depicted in Figure 1a. Because this function has two variables, its Hessian is a 2 by 2 matrix and there are three types of critical points: maximum (a peak), saddle (a valley), and a minimum. Observe that each level set of this function corresponds to a curve, and the topology of these curves change as the critical values are passed (Figure 1b). At each such value the associated level curve presents a singularity. In this case, at the level corresponding to the maxima, each curve is reduced to a point; and at the one corresponding to the saddle point, there is a self-intersecting point on the curve.

An analogous situation occurs for the function F , defined in $\mathbb{R}^3$. In this case, each level set is a closed surface immersed in $\mathbb{R}^3$. The

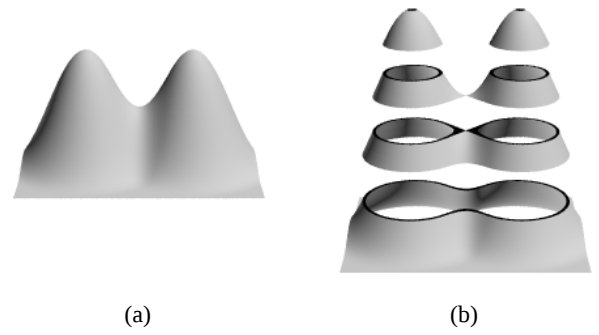


Figure 1: A two-variable scalar function show as a height field (a) and sliced to reveal some level sets (b).

topology of a given level set can be determined by comparing its threshold with the critical values of F . Therefore, the key point for guaranteeing a topologically correct polygonization of an implicit surface is to locate the critical points of its associated function.

Bottino et al. [4] generalized the *shrinkwrap* technique [15] to handle topology changes. For constructing a polygonal approximation for an implicit surface of level c_f , the method starts constructing a tessellation for the set associated with a lower level c_i , near zero. According to their implicit function definition, this level set is approximately a sphere with large radius, which can be easily triangulated. The method then proceeds by creating successive approximations for the intermediate level sets with the threshold incremented by $\Delta c > 0$, until the final value c_f is reached.

For each new level set, the mesh is updated by displacing the vertices on the surface and subdividing edges when the local curvature is high. To handle topological changes between two consecutive level sets, the technique searches for critical points in the neighborhood of each vertex. Also, to avoid unnecessary computations, a formula was proposed for trivially excluding the vertices whose neighborhood does not contain critical points. The remaining vertices are then employed as first guesses for Newton's method [16] to find possible critical points. After all nearby critical points are found, the appropriate topological change on the mesh is performed, based on the eigenvalues of the Hessian at those points.

Stander and Hart [18] proposed an alternative approach, named *inflation*, where the approximations are built as the initial threshold c_i decreases to a lower value c_f . The value c_i is chosen to be higher than any value given by the function F , so that the initial level set is empty. Differently from the modified shrinkwrap algorithm, all critical points are determined at the initial step. Their approximate sites are obtained by an interval subdivision algorithm, and then improved by an interval Newton's method.

While the threshold decreases, several critical values are passed, and the corresponding changes of topology are reflected in the polygonal mesh. The following sequence of critical points occurs: maximums, 2-saddles, 1-saddles, and minimums. This means that first the threshold will pass the critical value of a maximum point (a new component will be created). Then, it will pass the critical value of a 2-saddle (two components will merge) before it reaches the critical value of a 1-saddle (a hole is filled). Finally, it passes the critical value of a minimum (an air bubble is popped).

In [11], Hart et al. applied the previous technique on primitives with local support. Furthermore, for efficiency reasons, they used standard 3D Newton's method to improve the estimated sites of critical points.

3 Our Implicit Model

Our model consists of a collection of n unweighted spherical primitives in $\mathbb{R}^3$. The implicit function $F : \mathbb{R}^3 \rightarrow \mathbb{R}^+$ is given by

$$F(\mathbf{x}) = \sum_i^n f_i(\mathbf{x}), \quad (1)$$

where f_i is a real function associated with the primitive $\mathcal{P}_i$. This primitive is centered at the point $P_i(x_i, y_i, z_i)$, with positive influence radius r_i . We call a *region of influence* I_i of $\mathcal{P}_i$ the region in $\mathbb{R}^3$ where its influence is non-zero, that is,

$$I_i = \{\mathbf{x} \in \mathbb{R}^3 : \|\mathbf{x} - P_i\| < r_i\}.$$

The *common region of influence* $R_{j\dots k}$ is defined as the set of all points that have influence of the same primitives $\mathcal{P}_i$, where $i \in \{j, \dots, k\}$. In particular, when a primitive is isolated, $R_i = I_i$.

The real function f_i due to the primitive $\mathcal{P}_i$ is obtained by the composition of d with s_i :

$$f_i(\mathbf{x}) = d \circ s_i = d(s_i(\mathbf{x})),$$

where s_i and d are named, respectively, the shape and decay functions.

The *shape function* $s_i : \mathbb{R}^3 \rightarrow \mathbb{R}^+$ of $\mathcal{P}_i$ takes into consideration the associated influence radius, and gives the spherical appearance of the primitives:

$$s_i(\mathbf{x}) = \frac{\|\mathbf{x} - P_i\|}{r_i}. \quad (2)$$

The *decay function* controls the amount of influence given by a primitive as the distance from it increases. The function adopted was simultaneously proposed in [3] and [12], and is expressed by

$$d(t) = \begin{cases} (1 - t^2)^3, & \text{if } 0 \leq t \leq 1 \\ 0, & \text{otherwise} \end{cases}. \quad (3)$$

Note that d is invertible in the interval $[0, 1]$:

$$d^{-1}(t) = \sqrt{1 - \sqrt[3]{t}}.$$

From these definitions, our function f_i has some nice properties. It is monotonically decrescent as $\|\mathbf{x} - P_i\|$ increases in the influence region of $\mathcal{P}_i$, and vanishes when $\|\mathbf{x} - P_i\| \geq r_i$. Moreover, as it is C^2 continuous, this function provides smoother appearance than C^0 or C^1 continuous kernels (Figure 2).

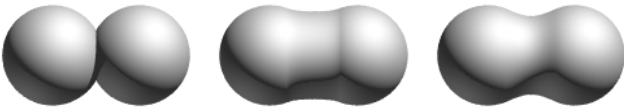


Figure 2: Two primitives with C^0 (left), C^1 (middle), and C^2 (right) kernels.

4 Estimates for Critical Points

In this section we describe an algorithm for finding a critical point due to the influence of n spherical primitives. Instead of isolating a critical point in a small enough refinement region [11], we step iteratively toward to it. In this way, we may avoid ambiguities (when there is more than one critical point in the same region) and numerical convergence problems.

4.1 Iterative Procedure

Consider n primitives $\mathcal{P}_i$. The critical points can be located by the following procedure:

1. (Maximums) For each primitive $\mathcal{P}_i$, with center P_i :

- Attribute initially the prediction $\hat{\mathbf{e}}_i = P_i$.
- Validate $\hat{\mathbf{e}}_i$, by verifying whether it is inside the influence region of primitives other than the ones already considered. If not, a maximum has been found and $\hat{\mathbf{x}}_i = \hat{\mathbf{e}}_i$.
- Update the prediction, by considering the contributions of all primitives that have non-zero influence at this point. Use as the seed for Newton's method the old $\hat{\mathbf{e}}_i$. Go to step 1b.

2. Construct an incidence graph: the vertices represent the maximums (and the corresponding primitives) and the edges, the common influence regions for each pair of maximums.

3. (2-saddles) For each edge ij in the graph:

- If a vertex, i or j , denotes a group of primitives, choose all pairs of primitives from both groups which have a common influence region.
- For each pair, compute the prediction $\hat{\mathbf{e}}_{ij}$ with Newton's method using as initial guess the point $\bar{P}_{ij}$ given by

$$\bar{P}_{ij} = \left(\frac{r_j}{r_i + r_j}\right)P_i + \left(\frac{r_i}{r_i + r_j}\right)P_j. \quad (4)$$

If the method fails or does not converge to a 2-saddle, a new guess is chosen on the basis of the previous one.

- Validate $\hat{\mathbf{e}}_{ij}$, by verifying whether it is inside the influence region of primitives other than the ones already considered. If not, a critical point has been found: make $\hat{\mathbf{x}}_{ij} = \hat{\mathbf{e}}_{ij}$ and go to step 3e.
- With Newton's method improve $\hat{\mathbf{e}}_{ij}$, by considering the contributions of all primitives that have non-zero influence at this point. Go to step 3c.
- Determine the classification of $\hat{\mathbf{x}}_{ij}$, on the basis of the eigenvalues of its Hessian.

4. Update the incidence graph by removing the edges between two vertices that have critical points of type other than a 2-saddle.

5. Assign as a *cycle* in the incidence graph each non-collinear and closed sequence of vertices whose corresponding primitives have a common influence region.

6. (1-saddles) For each cycle c in the graph:

- Compute the prediction $\hat{\mathbf{e}}_c$ using Newton's method. If the 2-saddles that surround the common region R_c are inside it, use their barycenter as the seed. Otherwise, replace the 2-saddles by points on the border of R_c and use the barycenter of them as the seed. If the method fails or does not converge to a 1-saddle, a new seed is chosen on the basis of the previous one.
- Validate $\hat{\mathbf{e}}_c$, by verifying whether it is inside the influence region of primitives other than the ones already considered. If not, a critical point has been found: make $\hat{\mathbf{x}}_c = \hat{\mathbf{e}}_c$ and go to step 6d.

- (c) With Newton's method improve $\hat{\mathbf{e}}_c$, by considering the contributions of all primitives that have non-zero influence at this point. Go to step 6b.
 - (d) Determine the classification of $\hat{\mathbf{x}}_c$, on the basis of the eigenvalues of its Hessian.
7. Assign as a *supercycle* in the incidence graph each set of cycles whose primitives build a non-coplanar configuration and have a common influence region.
 8. (Minimums) For each supercycle s in the graph:
 - (a) Compute the prediction $\hat{\mathbf{e}}_s$ using Newton's method. If the 1-saddles that surround the common region R_s are inside it, use their barycenter as the seed. Otherwise, replace the 1-saddles by points on the border of R_s and use the barycenter of them as the seed. If the method fails or does not converge to a minimum, choose a new seed on the basis of the previous one and repeat the procedure. Otherwise, make $\hat{\mathbf{x}}_s = \hat{\mathbf{e}}_s$.

9. End.

4.2 Analysis

We can perform the tracking of a critical point of F based on the fact that F is C^2 continuous.

When the influence regions do not overlap, we have in each region R_i only the individual contribution due to the primitive at the point $P_i = (x_i, y_i, z_i)$:

$$f_i(x, y, z) = \left(1 - \frac{(x - x_i)^2 + (y - y_i)^2 + (z - z_i)^2}{r_i^2}\right)^3.$$

The gradient $\nabla f_i(x, y, z)$ only vanishes at P_i . Because f_i is monotonically decrescent inside the influence radius r_i (as we move away from its center), P_i is the maximum of f_i in R_i .

The sum of two or more individual functions f_i , which are monotonically decrescent from their centers and whose centers are aligned, is either monotonically decrescent or "wavy". In the first case, the old maximums have collapsed into a new one (Figure 3a); in the second case, we have a 2-saddle between two maximums (Figure 3b). Note that we denote the group of primitives that collapse into a maximum with a single vertex in the incidence graph.

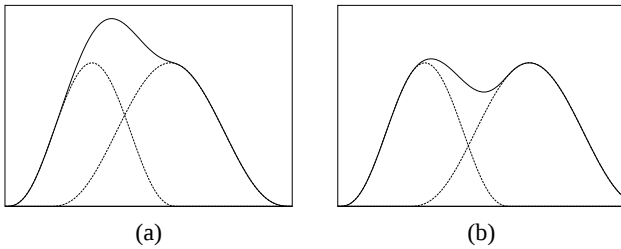


Figure 3: The sum of two monotonically decrescent functions.

To build a 2-saddle, two primitives $\mathcal{P}_i$ and $\mathcal{P}_j$ are sufficient, if they are placed in such a way that in their common influence region R_{ij} one of the functions is crescent and the other, decrescent. In this case, there is a point $\bar{P}_{ij}$ where

$$\left(1 - \frac{\|\bar{P}_{ij} - P_i\|^2}{r_i^2}\right)^3 = \left(1 - \frac{\|\bar{P}_{ij} - P_j\|^2}{r_j^2}\right)^3,$$

from which we obtain Eq. (4).

We observed that there is a point between $\bar{P}_{ij}$ and the center of the primitive of greater influence radius, where $f_i + f_j$ changes behavior, from monotonically decrescent ("descending" from P_i) to crescent ("climbing" towards P_j). This occurs along the direction $\bar{P}_i P_j$ (Figure 4). In a particular case, when $r_i = r_j$, this 2-saddle coincides with $\bar{P}_{ij}$. Hence, unless there are individual contributions in other directions, we only have two classes of critical points: maximums and 2-saddles, alternatively.

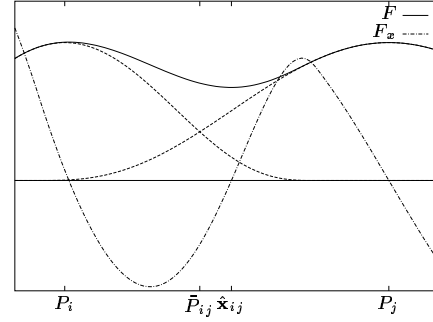


Figure 4: The 2-saddle between $\bar{P}_{ij}$ and the center of the primitive of greater influence radius, P_j .

This geometric insight was useful for choosing Newton's method for finding the 2-saddles, since there is at most one critical point, $\hat{\mathbf{x}}_{ij}$, between $\hat{\mathbf{x}}_i$ and $\hat{\mathbf{x}}_j$. Using $\bar{P}_{ij}$ as the first guess, the method should converge rapidly either to a maximum, $\hat{\mathbf{x}}_i$ or $\hat{\mathbf{x}}_j$, or to the 2-saddle, $\hat{\mathbf{x}}_{ij}$. When a maximum, say $\hat{\mathbf{x}}_i$, is sent as a solution, $\det H(\bar{P}_{ij})$ and $\det H(\hat{\mathbf{x}}_i)$ have the same sign (Figure 4). Then, we may pick

$$P_{ij}^1 = \frac{\bar{P}_{ij} + \hat{\mathbf{x}}_j}{2}$$

as another initial guess. In most cases, only two tries suffice. However, if this second point leads to $\hat{\mathbf{x}}_j$, one more try with

$$P_{ij}^2 = \frac{\bar{P}_{ij} + P_{ij}^1}{2}$$

should be sufficient to get a correct guess for $\hat{\mathbf{x}}_{ij}$. Figure 5 illustrates this procedure geometrically.

PSfrag replacements

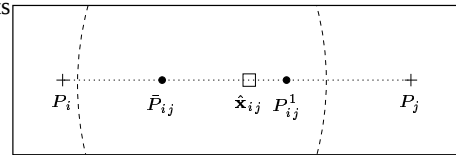


Figure 5: The improved search for the first guess of a 2-saddle.

One may argue that $\bar{P}_{ij}$ could lie near a maximum such that its gradient nearly vanishes and Newton's method fails. To remedy this situation, we use the midpoint of $\bar{P}_i P_j$ as another attempt.

A 1-saddle requires one second direction, λ , along which F also increases from this critical point. As our individual functions are monotonically decrescent from their centers, at least 3 non-collinear primitives, $\mathcal{P}_i$, $\mathcal{P}_j$, and $\mathcal{P}_k$, are necessary to create this valley configuration. The critical point must be surrounded by a "ring" of maximums and 2-saddles, such as a valley is surrounded by mountains and hills. Therefore, we look for 1-saddles only in the regions enclosed by maximums and 2-saddles. Analogously to the 2-saddles, we use as the initial estimate a point in the common influence regions R_{ijk} of those primitives.

In practice, we use the barycenter $\bar{P}_{ijk}$ of the surrounding 2-saddles, $\hat{x}_{ij}$, $\hat{x}_{jk}$, and $\hat{x}_{ik}$. If a 2-saddle is not inside R_{ijk} , say $\hat{x}_{ij}$, it is replaced by the point, $\hat{x}_{ij}^1$, on the border of R_k in the direction of its corresponding eigenvector. This ensures that we always choose a point with a “lower function value”. When $\det H(\hat{x}_{ijk})$ and $\det H(\bar{P}_{ijk})$ have different signs, Newton’s method will lead to a 2-saddle. We may, however, do another attempt with a new barycenter, after replacing the converged 2-saddle by the previous barycenter. This is done recursively until the method converges to the 1-saddle. In most cases, two tries are enough to get the correct initial estimate for $\hat{x}_{ijk}$. Figure 6 depicts graphically this idea.

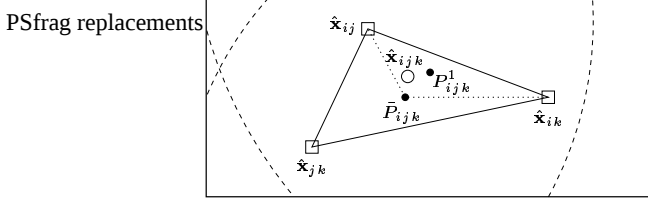


Figure 6: The improved search for the first guess of a 1-saddle.

We may generalize this procedure by supposing that n coplanar (but non-collinear) primitives build one closed sequence of maximums and 2-saddles. Then, we use as the initial estimate for the 1-saddle the barycenter of all 2-saddles or their corresponding points on the border of the common influence region. Because of the monotonic behavior of our function, there is only one critical point, namely 1-saddle, inside the common influence region of n primitives, if:

1. these primitives build one closed sequence of maximums and 2-saddles; and
2. the 2-saddles are either inside the common influence region or have corresponding points on the border of this region.

We can construct a minimum by imposing that F increases in any radial direction starting from it. Geometrically, this is equivalent to having in the neighborhood of the minimum “spatial valleys” that together build an “air bubble”. A minimal configuration that satisfies this requirement consists of 4 non-coplanar primitives. We search for a minimum in a similar way to what was done for a 1-saddle. It must be inside the common influence region of n primitives whose 1-saddles are not coplanar. Moreover, these saddles or their corresponding points on the border of the common influence region must belong to this region. We use the barycenter of these points as the initial estimate to the aimed minimum. If it fails, we may get a good first guess with a procedure similar to the one that we used to find out a correct estimate for an 1-saddle.

5 Examples

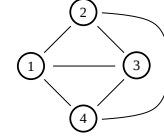
To illustrate the method proposed in Section 4.1, we present in this section some representative examples. The numerical computations were carried out in Mathematica [13] and the images were rendered by our customized version of the POV-Ray package [17].

Firstly, we consider a real function defined by four primitives with the same unitary influence radius, and with centers at $P_1(0.41,0.41,0.41)$, $P_2(0.41,-0.41,-0.41)$, $P_3(-0.41,0.41,-0.41)$, and $P_4(-0.41,-0.41,0.41)$. We show how the critical points can be obtained stepwise. The initial prediction (estimate) of each critical point ($\hat{x}$), the indices of primitives whose influence covers it (inf), and the number of iterations necessary to converge (it) are summarized in the following tables.

1. **Maximums:** The primitive centers were used as estimates for the maximums. Because there is no influence from other primitives, the maximums coincide with the centers, all having the same critical value 1.

i	estimate	inf	$\hat{x}_i$	it
1	P_1	1	(0.41,0.41,0.41)	0
2	P_2	2	(0.41,-0.41,-0.41)	0
3	P_3	3	(-0.41,0.41,-0.41)	0
4	P_4	4	(-0.41,-0.41,0.41)	0

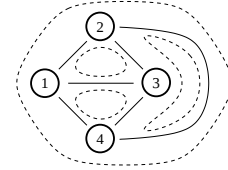
2. **Incidence graph:** To each primitive $i \in \{1, 2, 3, 4\}$ is associated one maximum. Therefore, we need four nodes to represent them. Due to the symmetric configuration, these primitives have mutual influences. This leads to the following incidence graph.



3. **2-saddles:** Eq. (4) was used to get the initial estimate for each pair of primitives that have a common influence region. Again, because of the symmetry of the configuration, all 2-saddles have the same critical value 0.5850.

ij	estimate	inf	$\hat{x}_{ij}$	it
12	$\bar{P}_{12}$	1,2	(0.41,0,0)	0
13	$\bar{P}_{13}$	1,3	(0,0,-0.41)	0
14	$\bar{P}_{14}$	1,4	(0,0.41,0)	0
23	$\bar{P}_{23}$	2,3	(0,0,0.41)	0
24	$\bar{P}_{24}$	2,4	(0,-0.41,0)	0
34	$\bar{P}_{34}$	3,4	(-0.41,0,0)	0

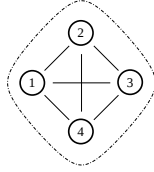
4. **Update the incidence graph:** As every critical point found in an influence region R_{ij} is a non-degenerate 2-saddle, no update on the incidence graph is necessary.
5. **Cycles in the graph:** In this case we have four closed sequences of nodes in our graph. They correspond, respectively, to the sets of primitives $\{1,2,3\}$, $\{1,3,4\}$, $\{2,3,4\}$, and $\{1,2,4\}$, as depicted in the graph below.



6. **1-saddles:** In this case, the barycenter of each group of three 2-saddles lies inside the common region of three primitives, R_{ijk} . Therefore, we used it as the first attempt to get the 1-saddles. Fortunately, Newton’s method converged to points of this type and no further estimations in the direction of eigenvectors were necessary. The corresponding critical value for each 1-saddle is 0.5056.

c	estimate	inf	$\hat{x}_c$	it
123	$(\hat{x}_{12} + \hat{x}_{23} + \hat{x}_{13})/3$	1..4	(0.125,0.125,-0.125)	2
124	$(\hat{x}_{12} + \hat{x}_{24} + \hat{x}_{14})/3$	1..4	(0.125,-0.125,0.125)	2
134	$(\hat{x}_{13} + \hat{x}_{34} + \hat{x}_{14})/3$	1..4	(-0.125,0.125,0.125)	2
234	$(\hat{x}_{23} + \hat{x}_{34} + \hat{x}_{24})/3$	1..4	(-0.125,-0.125,-0.125)	2

7. **Supercycle in the graph:** Only with the primitives of all four cycles we were able to define a minimal non-coplanar configuration. Therefore, we group these four cycles into one supercycle, as shown below.



8. **Minimums:** Again, because of the symmetry, we can use as estimate for the minimum the barycenter of the four previously found 1-saddles. Its critical value is 0.4872.

s	estimate	$\hat{\mathbf{x}}_s$	it
1234	$(\hat{\mathbf{x}}_{123} + \hat{\mathbf{x}}_{124} + \hat{\mathbf{x}}_{134} + \hat{\mathbf{x}}_{234})/4$	(0,0,0)	0

9. End.

From the critical values found above, it is easy to characterize the topology of the implicit surface S_c for distinct values of c . As there are four distinct critical values, five topological configurations can be characterized (Figure 7).

no components	$c > 1$
4 components	$0.5850 < c \leq 1$
1 component with 4 holes	$0.5056 < c \leq 0.5850$
1 component with 1 bubble	$0.4872 < c \leq 0.5056$
1 filled component	$0 < c \leq 0.4872$

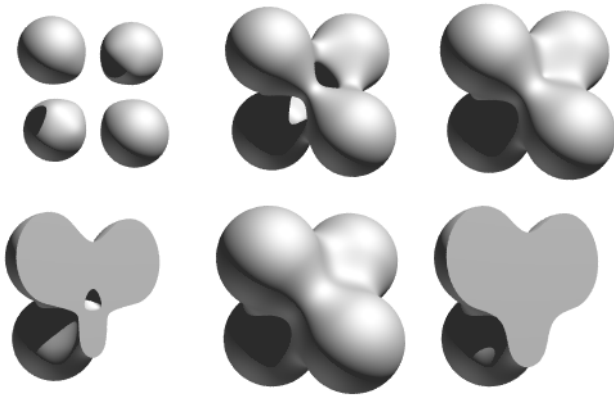


Figure 7: Symmetric configuration: $c=0.7$ (top left), $c=0.55$ (top middle), $c=0.5$ (top right) and slice (bottom left), and $c=0.45$ (bottom middle) and slice (bottom right).

To illustrate a situation where Newton's method may fail with the initial estimate given by Eq. (4), we analyzed a function defined by two primitives with influence radius $r_1=2$ and $r_2=3$. They are centered, respectively, at $P_1(0,0,0)$ and $P_2(2,1,0)$. Although the method diverges with our first estimate, we knew that there must be a 2-saddle between two maximums from the existence of the common influence region R_{12} . Then, an attempt with a new $\hat{P}_{ij}$ was carried out. In the second try, we were able to locate the 2-saddle after 2 iterations. Figure 8 shows the implicit surface associated with $c=0.9138$. Note that there is a singularity in the surface that corresponds to the 2-saddle found in F . The next table summarizes the critical points found.

point	class	position	crit. value	it
$\hat{\mathbf{x}}_1$	maximum	(0.278,0.139,0)	1.1327	2
$\hat{\mathbf{x}}_2$	maximum	(2,1,0)	1	0
$\hat{\mathbf{x}}_{12}$	2-saddle	(1.338,0.669,0)	0.9138	2

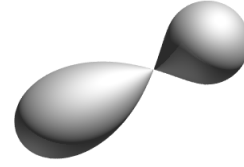


Figure 8: A two primitive configuration, with $c=0.9138$.

Another interesting example is shown in Figure 9. It contains a shared maximum created by two primitives. It also shows that even when there are common regions of influence of three primitives, the existence of a 1-saddle cannot be assured. Moreover, this example illustrates that not always we have a 2-saddle between a pair of primitives. The parameters for this configuration are: $r_1=2$, $r_2=2.6$, $r_3=3.5$, $r_4=1.5$, $P_1(0,0,0)$, $P_2(4,0,0)$, $P_3(2,1,0)$, and $P_4(2,0.5,1.5)$. The table below lists the critical points obtained with our method. Note that the critical points $\hat{\mathbf{x}}_{12}$ and $\hat{\mathbf{x}}_{14}$ do not exist, due to the influence of P_3 .

point	class	position	crit. value	it
$\hat{\mathbf{x}}_4$	maximum	(2.007,0.557,1.317)	1.5508	2
$\hat{\mathbf{x}}_2 \equiv \hat{\mathbf{x}}_3$	maximum	(2.445,0.278,0)	1.3260	3
$\hat{\mathbf{x}}_1$	maximum	(0.290,0.145,0)	1.2686	2
$\hat{\mathbf{x}}_{24} \equiv \hat{\mathbf{x}}_{34}$	2-saddle	(2.618,0.526,0.496)	1.1234	4
$\hat{\mathbf{x}}_{13}$	2-saddle	(1.454,0.727,0)	0.9506	4

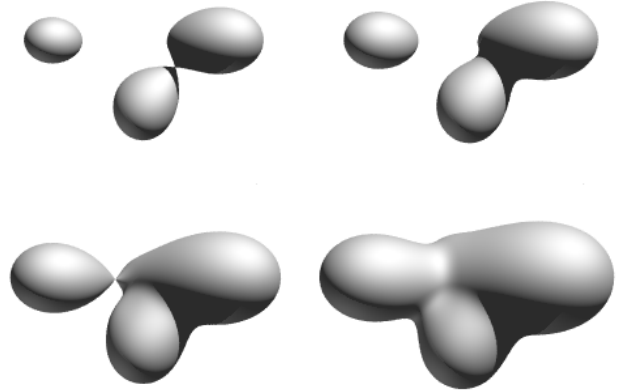


Figure 9: A four primitive configuration, with $c=1.1234$ (top left), $c=1.05$ (top right), $c=0.9506$ (bottom left), and $c=0.8$ (bottom right).

Figure 10 exemplifies a situation where only one maximum is associated to several primitives: four primitives with radius $r_1=1$, $r_2=0.5$, $r_3=0.4$, and $r_4=0.4$, centered at $P_1(0,0,0)$, $P_2(0.6,0,0)$, $P_3(0.4,-0.21,0)$, and $P_4(0.4,0.21,0)$. In this case, there are three primitives that build a single maximum. Hence, our incidence graph is constituted by two nodes. One corresponds to a "enlarged" primitive and the other to P_1 . Observe that in this case we could successfully distinguish two critical points that are very close, as attested in the following table. Also, as expected, the 2-saddles and maximums surround a 1-saddle.

point	class	position	crit. value	it
$\hat{\mathbf{x}}_2 \equiv \hat{\mathbf{x}}_3 \equiv \hat{\mathbf{x}}_4$	maximum	(0.4367,0,0)	1.9770	8
$\hat{\mathbf{x}}_1$	maximum	(0,0,0)	1	0
$\hat{\mathbf{x}}_{13}$	2-saddle	(0.0841,-0.044,0)	0.9817	3
$\hat{\mathbf{x}}_{24}$	2-saddle	(0.0841,0.044,0)	0.9817	3
$\hat{\mathbf{x}}_{123} \equiv \hat{\mathbf{x}}_{124}$	1-saddle	(0.100,0,0)	0.9788	5

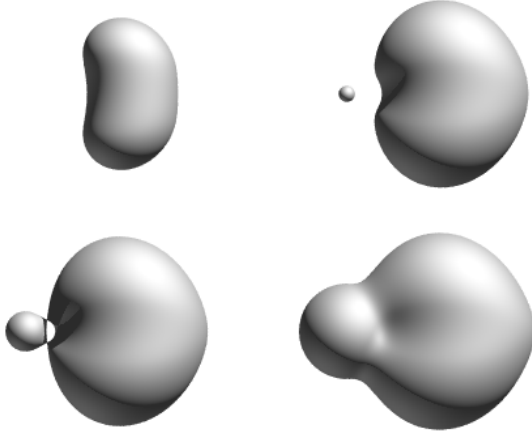


Figure 10: A four primitive “valley” configuration, with $c=1.4$ (top left), $c=0.997$ (top right), $c=0.9817$ (bottom left), and $c=0.9$ (bottom right).

We present in Figure 11 a configuration made of six primitives with the same unitary influence radius. The centers are $P_1(1.1,0,0)$, $P_2(0.55,0.9526,0)$, $P_3(-0.55,0.9526,0)$, $P_4(-1.1,0,0)$, $P_5(-0.55,-0.9526,0)$, and $P_6(0.55,-0.9526,0)$, building a ring. It is interesting to note that there is no 1-saddle, although the maximums and 2-saddles build a perfect cycle in $\mathbb{R}^3$. This is expected, as there is no common influence region for these six primitives.

point	class	position	crit. value	it
$\hat{\mathbf{x}}_1$	maximum	(1.1,0,0)	1	0
$\hat{\mathbf{x}}_2$	maximum	(0.55,0.9526,0)	1	0
$\hat{\mathbf{x}}_3$	maximum	(-0.55,0.9526,0)	1	0
$\hat{\mathbf{x}}_4$	maximum	(-1.1,0,0)	1	0
$\hat{\mathbf{x}}_5$	maximum	(-0.55,-0.9526,0)	1	0
$\hat{\mathbf{x}}_6$	maximum	(0.55,-0.9526,0)	1	0
$\hat{\mathbf{x}}_{12}$	2-saddle	(0.825,0.476,0)	0.6787	0
$\hat{\mathbf{x}}_{23}$	2-saddle	(0,0.953,0)	0.6787	0
$\hat{\mathbf{x}}_{34}$	2-saddle	(-0.825,0.476,0)	0.6787	0
$\hat{\mathbf{x}}_{45}$	2-saddle	(-0.825,-0.476,0)	0.6787	0
$\hat{\mathbf{x}}_{56}$	2-saddle	(0,-0.953,0)	0.6787	0
$\hat{\mathbf{x}}_{16}$	2-saddle	(0.825,-0.476,0)	0.6787	0

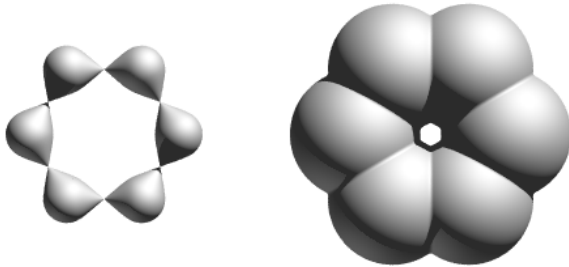


Figure 11: A six primitive configuration, with $c=0.6787$ (left) and $c=0.001$ (right).

Finally, we show a more involved example in Figure 12. The female torso shown on the top is made of 22 primitives, with radii varying from 0.9 to 7.5 units and with threshold equal to 0.9. The positions of these primitives are presented in the middle of the figure. Finally, with our proposed procedure we could successfully determine the critical points and generate the interesting surfaces presented at the bottom.

point	class	position	crit. value	it
$\hat{\mathbf{x}}_Q$	maximum	(0,6.435,-0.708)	3.2530	8
$\hat{\mathbf{x}}_L$	maximum	(-1.382,6.318,0.057)	3.1774	8
$\hat{\mathbf{x}}_M$	maximum	(1.382,6.318,0.057)	3.1774	8
$\hat{\mathbf{x}}_P$	2-saddle	(-1.003,6.359,-0.245)	3.1690	7
$\hat{\mathbf{x}}_R$	2-saddle	(1.003,6.359,-0.245)	3.1690	4
$\hat{\mathbf{x}}_S$	maximum	(-2.322,6.036,1.607)	2.8077	3
$\hat{\mathbf{x}}_T$	maximum	(2.322,6.036,1.607)	2.8077	3
$\hat{\mathbf{x}}_E$	maximum	(-2.150,-2.354,-1.713)	2.6602	3
$\hat{\mathbf{x}}_F$	maximum	(2.150,-2.354,-1.713)	2.6602	3
$\hat{\mathbf{x}}_U$	2-saddle	(-2.146,6.113,1.212)	2.6293	4
$\hat{\mathbf{x}}_V$	2-saddle	(2.146,6.113,1.212)	2.6293	4
$\hat{\mathbf{x}}_{GH}$	2-saddle	(0,-1.376,-1.613)	1.7683	3
$\hat{\mathbf{x}}_I$	2-saddle	(0,1.268,-0.778)	1.6248	2

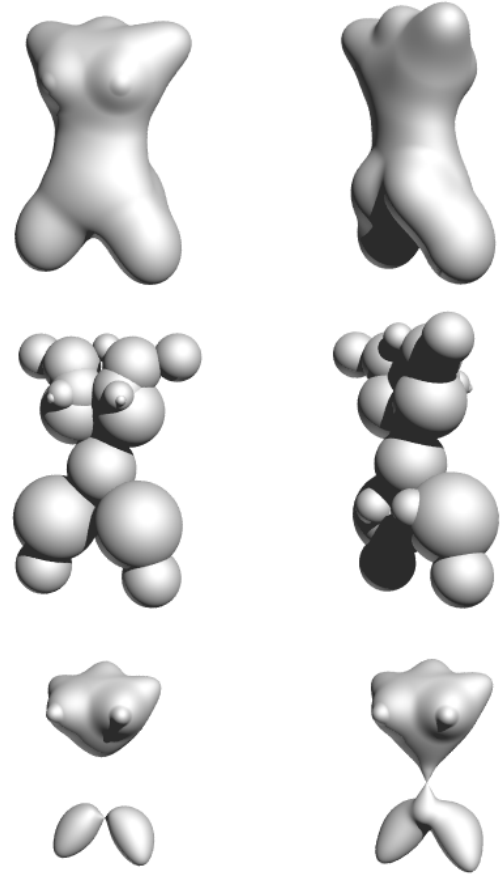


Figure 12: A female torso, with $c=0.9$ (top left and top right), $c=1.7683$ (bottom left), and $c=1.6244$ (bottom right).

6 Conclusions

We presented an algorithm that determines efficiently good first guesses for the critical points of a skeleton-based implicit model.

Under the constraints that the primitives are unweighted polynomial functions with monotonically decreasing contributions, we showed that a tight initial guess may be achieved iteratively by analyzing the positional relationship among the primitives. In our numerical experimentation with these estimates, we realized that Newton's method converges in less than four iterations in most situations. Furthermore, once we look for the critical points directly in their neighborhood, we could overcome the problem of "false critical points" reported in [11], without any numerical assumption.

Although our algorithm was conceived for a class of unweighted primitive functions, we believe that it also works well for positive weights. Slight modifications to Eq. (4) may be necessary to take these weights into consideration. The refining search procedure that leads to predictions for critical points should be the same, since the primitive functions have similar behavior.

Instead of the proposed recursive procedure for finding the initial guess in the common influence region, one may use an interval search. It should be interesting to compare the performance of these two approaches.

As further work, we plan to integrate the proposed procedure for finding critical points into the polygonization procedure proposed by Desbrun et al. and Crespín et al., in order to guarantee topological correctness of the resulting mesh.

Acknowledgments

We wish to thank the referees for their helpful suggestions on the improvement of this paper. We also thank CNPq for granting the financial support for this work.

References

- [1] James F. Blinn. A Generalization of Algebraic Surface Drawing. *ACM Transactions on Graphics*, 1(3):235–256, July 1982.
- [2] Jules Bloomenthal and Brian Wyvill. Interactive Techniques for Implicit Modeling. *Computer Graphics (1990 Symposium on Interactive 3D Graphics)*, 24(2):109–116, March 1990.
- [3] Jules Bloomenthal (ed). *Introduction to Implicit Surfaces*, Morgan Kaufmann, 173–175, 1997.
- [4] Andrea Bottino, Wim Nuij, and Kees van Overveld. How to Shrinkwrap through a Critical Point: an Algorithm for the Adaptive Triangulation of Iso-Surfaces with Arbitrary Topology. *Proceedings of Implicit Surfaces '96*, 53–73, October 1996.
- [5] Benôit Crespín, Pascal Guitton, and Christophe Schlick. Efficient and Accurate Tessellation of Implicit Sweep Objects. *Constructive Solid Geometry '98*, April 1998.
- [6] Mathieu Desbrun, Nicolas Tsingos, and Marie-Paule Gascuel. Adaptive Sampling of Implicit Surfaces for Interactive Modeling and Animation. *Proceedings of Implicit Surfaces '95*, 171–185, April 1995.
- [7] Marie-Paule Cani-Gascuel and Mathieu Desbrun. Animation of Deformable Models using Implicit Surfaces. *IEEE Transactions on Visualization and Computer*, 3(1):39–50, January 1997.
- [8] Luiz Henrique de Figueiredo, Jonas Gomes, Demetri Terzopoulos, and Luiz Velho. Physically-Based Methods for Polygonization of Implicit Surfaces. *Proceedings of Graphics Interface '92*, 250–257, May 1992.
- [9] Luiz Henrique de Figueiredo and Jonas Gomes. Sampling implicit objects with physically-based particle systems. *Computers and Graphics*, 20(3):365–375, May 1996.
- [10] John C. Hart. Morse Theory for Implicit Surface Modeling. *Proceedings of Visualization and Mathematics '97*, 1997.
- [11] John C. Hart, Antoine Durr, and Douglas Harsh. Critical Points of Polynomial Metaballs. *Proceedings of Implicit Surfaces '98*, 69–76, June 1998.
- [12] Marcelo de G. Malheiros and Wu, Shin – Ting. A Hierarchical Skeleton-based Implicit Model. *Proceedings of SIBGRAP '97 (Brazilian Symposium on Computer Graphics and Image Processing)*, 65–70, October 1997.
- [13] Wolfram Research, Inc. <http://www.wolfram.com/>
- [14] John Milnor. Morse Theory. *Annals of Mathematics Studies*, vol. 51, Princeton University Press, 1963.
- [15] C.W.A.M. van Overveld and Brian Wyvill. Shrinkwrap: An adaptive algorithm for polygonizing an implicit surface. *Tech. Rep. 93/514/19*, University of Calgary, March 1993.
- [16] William H. Press, Saul A. Teukolsky, William T. Vetterling, and Brian P. Flannery. *Numerical Recipes in C*, Cambridge University Press, 1992.
- [17] POV-Ray Team. <http://www.povray.org/>
- [18] Barton T. Stander and John C. Hart. Guaranteeing the Topology of an Implicit Surface Polygonization for Interactive Modeling. *SIGGRAPH 97 Conference Proceedings*, 279–286, August 1997.
- [19] Greg Turk. Generating Textures on Arbitrary Surfaces Using Reaction-Diffusion. *SIGGRAPH 91 Conference Proceedings*, 25(4):289–298, July 1991.
- [20] Andrew P. Witkin and Paul S. Heckbert. Using Particles to Sample and Control Implicit Surfaces. *SIGGRAPH 94 Conference Proceedings*, 269–278, July 1994.
- [21] Geoff Wyvill, Craig McPheeters, and Brian Wyvill. Data structure for soft objects. *The Visual Computer*, 2(4):227–234, August 1986.