LAB REPORT: LAB 4

TNM079, MODELING AND ANIMATION

Algot Sandahl
algsa119@student.liu.se

Wednesday 16th June, 2021

Abstract

This lab report covers implicit surfaces and their differential properties. The many forms of quadric surfaces are discussed. Furtermore some constructive solid geometry techniques such as the union, intersection, and difference operators, as well as super elliptic blending techniques are discussed.

The implicit representation provides interesting properties and the constructive solid geometry operators together with for example the quadric surfaces enable quite complex shapes.

1 Background

Sometimes it can be beneficial to represent surfaces in terms of mathematical functions instead of as explicit triangle meshes. One way of doing so would be to let one dimension be a function of the others:

$$z = f(x, y). \tag{1}$$

This is called an explicit representation.

Another way of representing surfaces by mathematical functions is the implicit surface representation—the topic of this lab. Let a function

$$f(\mathbf{x}) \to \eta, \quad \mathbf{x} \in \mathbb{R}^3, \quad \eta \in \mathbb{R}, \tag{2}$$

i.e. map a 3-dimensional vector $\mathbf{x} = [x, y, z]$ to a scalar. This function can be used to define a surface in three dimensions by defining the surface as all points where f has the same value:

$$\mathcal{S}(C) = \Big\{ \{\mathbf{x}\} : f(\mathbf{x}) = C \Big\} \tag{3}$$

where C is the iso-value. In computer graphics, the iso-value is usually set to zero—this way the sign of f can be used to determine if a point is inside or outside the surface. Using this convention, f can be defined such that

$$f(\mathbf{x}) \begin{cases} < 0, & \text{if } \mathbf{x} \text{ is inside the surface} \\ = 0, & \text{if } \mathbf{x} \text{ is on the surface} \\ > 0, & \text{if } \mathbf{x} \text{ is outside the surface} \end{cases} \tag{4}$$

To render the implicit surface using a traditional graphics pipeline, the function is sampled in a grid, with the surface given as the interface between the negative and positive values. For ray tracing, there are often better alternatives. For example ray intersections with quadric surfaces can be easily solved without sampling the function.

For shading, surface normals are of utmost importance. The normal of an implicit surface is the normalized gradient

$$\nabla f(\mathbf{x}) = \begin{bmatrix} \frac{\partial f(\mathbf{x})}{\partial x} & \frac{\partial f(\mathbf{x})}{\partial y} & \frac{\partial f(\mathbf{x})}{\partial z} \end{bmatrix}^T. \tag{5}$$

As known, the partial derivative, with respect to x, at a point, $\mathbf{x}_0$, is defined as

$$\frac{\partial f(\mathbf{x}_0)}{\partial x} = \lim_{h \to 0} \frac{f(\mathbf{x}_0 + h\hat{\mathbf{e}}_x) - f(\mathbf{x}_0)}{h}. \tag{6}$$

When calculating the derivative in applications, h can not be arbitrarily small. The derivative can however be approximated by a finite difference by choosing a small enough constant, ε, instead of $h \to 0$, resulting in a forward difference scheme. Nonetheless, it is often more accurate to evaluate the derivative as a central difference

$$D_x(\mathbf{x}_0) = \frac{f(\mathbf{x}_0 + \varepsilon\hat{\mathbf{e}}_x) - f(\mathbf{x}_0 - \varepsilon\hat{\mathbf{e}}_x)}{2\epsilon} \tag{7}$$

which approximates the derivative in a symmetric way. D_y and D_z are naturally calculated in the same way as D_x. Then the gradient can be calculated as

$$\nabla f(\mathbf{x}) \approx \begin{bmatrix} D_x(\mathbf{x}) & D_y(\mathbf{x}) & D_z(\mathbf{x}) \end{bmatrix}^T \tag{8}$$

and the surface normal for a point $\mathbf{x}$ on the surface can be calculated as

$$\hat{\mathbf{n}}(\mathbf{x}) = \frac{\nabla f(\mathbf{x})}{\|\nabla f(\mathbf{x})\|} \tag{9}$$

One type of implicit surface is the quadric surface, and its function is defined as

$$\begin{aligned} f(x, y, z) &= Ax^2 + 2Bxy + 2Cxz \\ &\quad + 2Dx + Ey^2 + 2Fyz \\ &\quad + 2Gy + Hz^2 + 2Iz + J \\ &= \mathbf{p}^T\mathbf{Q}\mathbf{p} \end{aligned} \tag{10}$$

where $\mathbf{p}^T = [x, y, z, 1]$ and the matrix is

$$\mathbf{Q} = \begin{bmatrix} A & B & C & D \\ B & E & F & G \\ C & F & H & I \\ D & G & I & J \end{bmatrix}. \tag{11}$$

The matrix form $\mathbf{p}^T\mathbf{Q}\mathbf{p} = 0$ in particular, is convenient for computer graphics. The quadric in its general form is quite flexible, and by enforcing certain constraints on the values of $\mathbf{Q}$, it can describe a lot of different shapes. Some examples of surfaces that can be represented as a quadric are listed below.

- Planes: $ax + by + cz = 0$
- Cylinders: $x^2 + y^2 - 1 = 0$
- Ellipsoids: $\frac{x^2}{a^2} + \frac{y^2}{b^2} + \frac{z^2}{c^2} - 1 = 0$
- Cones: $x^2 + y^2 - z^2 = 0$
- Paraboloids: $x^2 \pm y^2 - z = 0$
- Hyperboloids: $x^2 + y^2 - z^2 \pm 1 = 0$

What values should be in the matrix $\mathbf{Q}$ for the different surface types can be easily seen by comparing the equations with (10). For example, the plane gives $2D = a$, $2G = b$, $2I = c$, and the rest of the elements are zero. Thus the quadric matrix of the plane, $ax + by + cz = 0$, is

$$\mathbf{Q}_{\text{plane}} = \begin{bmatrix} 0 & 0 & 0 & a/2 \\ 0 & 0 & 0 & b/2 \\ 0 & 0 & 0 & c/2 \\ a/2 & b/2 & c/2 & 0 \end{bmatrix} \tag{12}$$

The analytic expression of a quadric's gradient is known and can be written as

$$\nabla f(x, y, z) = 2\mathbf{Q}_{\text{sub}}\mathbf{p} \tag{13}$$

where $\mathbf{Q}_{\text{sub}} \in \mathbb{R}^{3\times4}$ is the first three rows of the quadric matrix $\mathbf{Q}$. This expression is well-suited for calculations, and eliminates the need for the approximate gradient given by (8) when working with quadric surfaces.

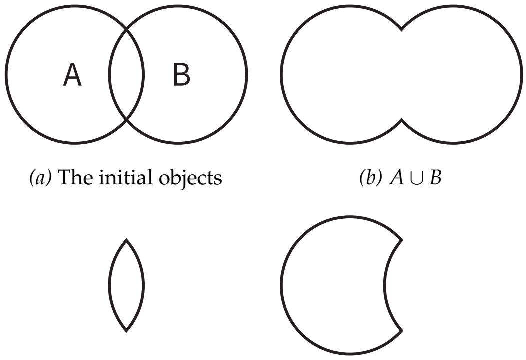
On their own, simple implicit functions are usually not all that interesting, but many objects can be combined using boolean operations in what is known as constructive solid geometry, to achieve more complex shapes. The boolean operations union, intersection, and difference, between two implicit surfaces, A and B, can be calculated as

$$f_{A\cup B}(\mathbf{x}) = \min\Big(f_A(\mathbf{x}), f_B(\mathbf{x})\Big) \tag{14}$$

$$f_{A\cap B}(\mathbf{x}) = \max\Big(f_A(\mathbf{x}), f_B(\mathbf{x})\Big) \tag{15}$$

$$f_{A-B}(\mathbf{x}) = \max\Big(f_A(\mathbf{x}), -f_B(\mathbf{x})\Big) \tag{16}$$

The intuition for this can be realized by observing Figure 1. Take for example the intersection operator—by taking the maximum of f_A and f_B, the result is negative only at points where both f_A and f_B are negative, i.e. at points that are inside both A and B. The result is positive for points where at least one of the functions



(a) The initial objects

(b) $A \cup B$

(c) $A \cap B$

(d) $A - B$

Figure 1: The boolean operations illustrated in 2D.

are positive. The surface is, as defined, the interface between these regions, i.e. where the value is zero. Similar reasoning can be applied to the other operators.

While the edges created by the boolean operations are continuous, their derivatives are not; this creates problems when defining the surface normal at these points. It would be beneficial to be able to “smoothen” the surface at these points. The following paragraph explains how this can be done.

The surface representation can be transformed to a density function

$$D_A(\mathbf{x}) = \exp(-f_A(\mathbf{x})). \tag{17}$$

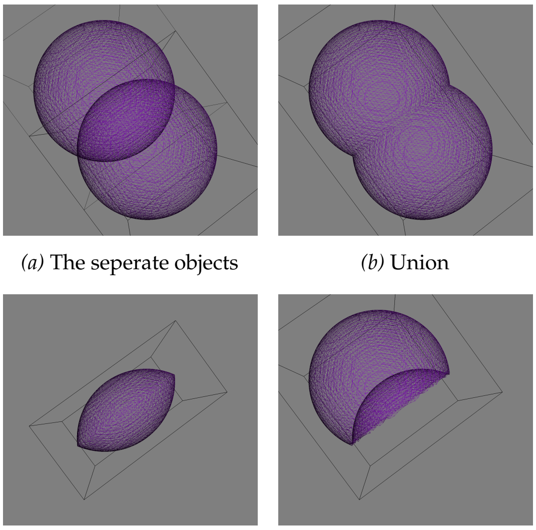
As a consequence of the way the implicit surface function is defined, the inside and outside of the surface represented as a density function is defined as

$$D_A(\mathbf{x}) \begin{cases} > 1, & \text{if } \mathbf{x} \text{ is inside the surface} \\ = 1, & \text{if } \mathbf{x} \text{ is on the surface} \\ \in [0, 1), & \text{if } \mathbf{x} \text{ is outside the surface} \end{cases} \tag{18}$$

To obtain the implicit surface from the density function, the inverse of (17) is used:

$$f_A(\mathbf{x}) = -\ln D_A(\mathbf{x}). \tag{19}$$

With this knowledge, operations can be performed on the density function as opposed to the implicit surface function, $f(\mathbf{x})$. An interesting application of this is the super-elliptic blending—a “smoother” version of the



(a) The seperate objects

(b) Union

(c) Intersection

(d) Difference

Figure 2: The results of the boolean operators on two spheres shown as wireframes.

boolean operations. The density functions for super-elliptic blending operations at a point are defined by

$$D_{A\cup B} = \left(D_A^p + D_B^p\right)^{1/p} \tag{20}$$

$$D_{A\cap B} = \left(D_A^{-p} + D_B^{-p}\right)^{-1/p} \tag{21}$$

$$D_{A-B} = \left(D_A^{-p} + D_{-B}^{-p}\right)^{-1/p} \tag{22}$$

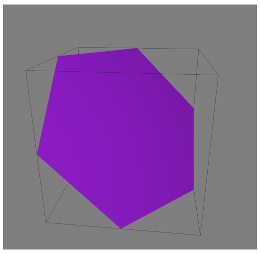
As p approaches infinity, the results of these operations approach the plain boolean operations defined in (14)–(16). Smaller p produces smoother edges.

2 Results

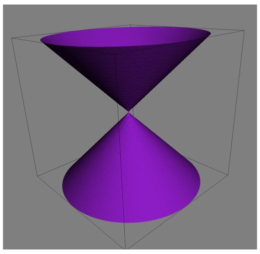
This section presents the results of the lab.

2.1 CSG Operators

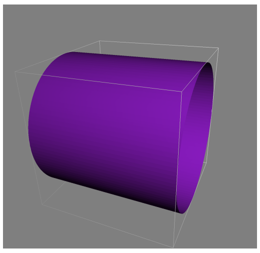
The results of applying the boolean operations to two spheres is shown in Figure 2. Note the sharp edges between the different parts.



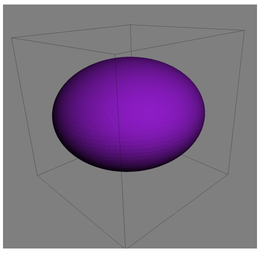
(a) Plane



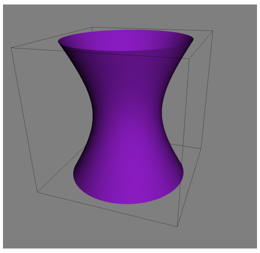
(b) Cone



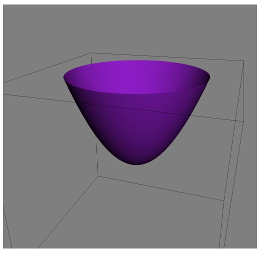
(c) Cylinder



(d) Ellipsoid



(e) Hyperboloid



(f) Paraboloid

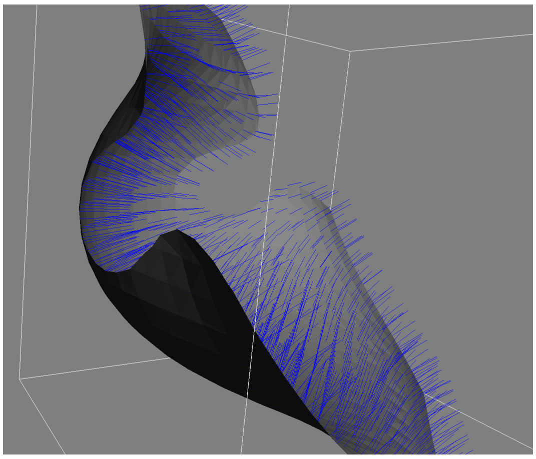
Figure 3: Different quadric surfaces. Note that the surfaces are only sampled inside the box.

2.2 Quadric Surfaces

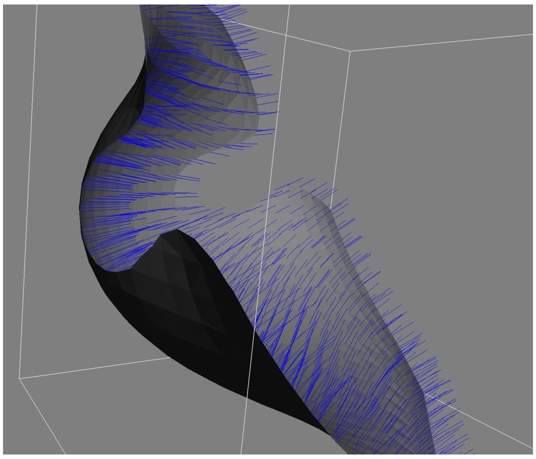
Figure 3 shows some examples of wildly different surfaces that are all represented by quadrics. As a consequence of the rendering technique, the surfaces are only rendered inside a bounding box. Most of the surfaces do however extend infinitely.

2.3 Discrete Gradient Operator

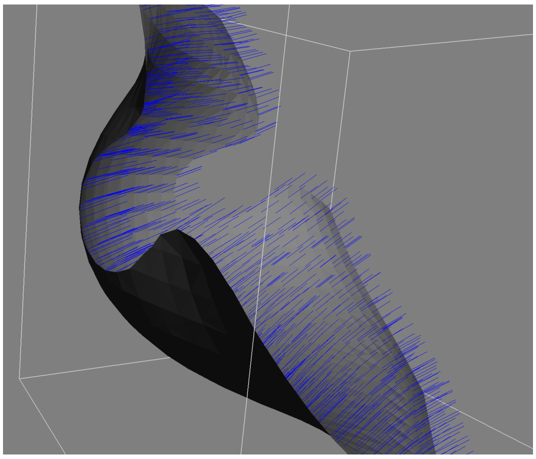
Figure 4 shows the effects of the step size ε on the discrete gradient operator. As seen a smaller ε produces a gradient that follows the curvature of the surface, but larger steps produces gradients that are almost parallel over the whole surface.



(a) $\varepsilon = 0.01$

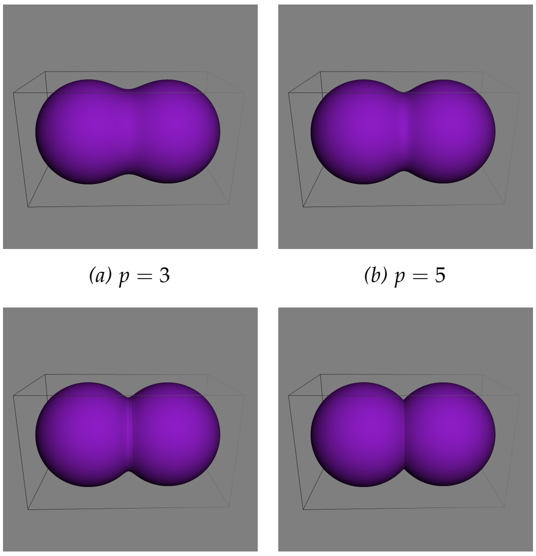


(b) $\varepsilon = 0.51$



(c) $\varepsilon = 1$

Figure 4: The discrete gradient operator (8) using different ε.



(a) $p = 3$ (b) $p = 5$ (c) $p = 10$ (d) $p = 100$

Figure 5: Super-elliptic blending of two spheres using different values for p.

2.4 Super-Elliptic Blending

The results of applying super-elliptic blending with the union operator for two spheres are shown in Figure 5. Different values for the blending parameter, p, are compared to demonstrate its effect. The super-elliptic intersection and difference operators produced smoother edges in the same way as the super-elliptic union operator, albeit on convex edges.

3 Conclusion

Representing surfaces by implicit functions provides some interesting properties. As opposed to a triangle mesh, the implicit representation provides a representation with infinite resolution. For a traditional hardware graphics pipeline, however, the surface has to be triangulated before rendering, somewhat negating the usefulness of this property. It does however still mean that the full resolution representation can be sampled to a suitable resolution when needed.

3.1 CSG Operators

The boolean operations provide a simple but powerful way to make interesting compound objects from simpler objects. A drawback is that sharp edges are created, which can be unwanted.

3.2 Quadric Surfaces

As seen, quadric surfaces can represent a wide variety of different surfaces. The shapes are undeniably simple but when combined with the CSG operators, many interesting shapes can be represented.

The quadric surfaces also provide a useful way of calculating the gradient, that does not depend on any finite difference method.

3.3 Discrete Gradient Operator

The smaller step sizes produce better results. As ε decreases, the gradient is evaluated in an increasingly localized area around the point, which provides a better result. When ε is large, the two points, $\mathbf{x}_0 + \varepsilon\hat{\mathbf{e}}_x$ and $\mathbf{x}_0 - \varepsilon\hat{\mathbf{e}}_x$, are both located far away from the point $\mathbf{x}_0$ where the gradient is evaluated—this means that local changes are not catched.

3.4 Super-Elliptic Blending

As shown, the super-elliptic blending is an effective method for performing boolean operations without the sharp edges. It is clear that smaller values for p produce smoother edges, and larger p produces sharper edges, but other than that it can be hard to know exactly what the resulting shape will be.

Lab Partner and Grade

The lab was done together with Viktor Sjögren. All lab tasks marked 3, 4, or 5b were finished, and the report aims for grade 5.