LAB REPORT: LAB 4

TNM079, MODELING AND ANIMATION

Alma Linder
almli825@student.liu.se

Friday 25th April, 2025

Abstract

This lab work is focused on implicit surfaces and showcases the efficiency with which these surfaces can be merged or cut, and the mathematical simplicity to find their real normals and gradients based on the innate attributes of the surface definition itself. Different surfaces from the quadric implicit surface function are furthermore derived.

The boolean operators for merging and cutting surfaces prove that modeling new surfaces and shapes is possible from several initial ones. Sampling distance experiments show increasingly smoothened iso-surfaces. Some of the surfaces derived from the quadric function exemplify both the simplicity of defining a diversity of shapes, and the correct normals and gradients along these surfaces.

This laboratory work aims for `grade 3`.

1 Background

Implicit surfaces (or iso-surfaces) - unlike triangle meshes - are described by continuous functions $F(x, y, z)$ (in 3D space). Since the function F is defined in all points in space due to continuity, the surface will be manifold and exist infinitely in space. By adding the condition $F(x, y, z) = C$ with $C = 0$ (C designates the iso-value) one surface to work with is extracted. Thus, varying the value of C will extract different surfaces from the space in which F in continuous. Figure 1 illustrates this concept for a sphere.

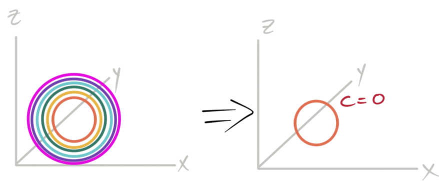


Figure 1: Implicit surfaces infinitely defined as contours from a function $F(x, y, z) = C$, where the iso value C is a number.

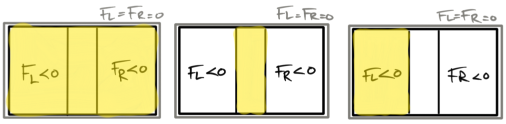
Implementation-wise this means iso-surfaces and points on/around them were handled as triple function values (x, y, z) of F only, rather than structures of triangle components as for triangular half-edge meshes.

Figure 1 also shows that points on the surface will give $C = 0$, points inside the surface result in $C < 0$ and points outside the surface yield $C > 0$. This knowledge can be used for both operations on iso-surface combinations - namely constructive solid geometry (CSG) - and sampling.

Firstly, turning to CSGs, the above concept implies that the union of two iso-surfaces L and R can be computed according to Equation 1a. Similarly, Equations 1b and 1c hold too, for intersection and difference respectively.

$$F_{L\cup R} = \min(F_L, F_R) \tag{1a}$$
$$F_{L\cap R} = \max(F_L, F_R) \tag{1b}$$
$$F_{L-R} = \max(F_L, -F_R) \tag{1c}$$

Figure 2 shows why the equations in 1 are valid boolean operators: If F_L and F_R are the implicit functions for the surfaces L and R, then the minimum $F_L(\chi)$ and $F_R(\chi)$ for a point χ will be negative if χ lies inside either shape, and thus represents the union $F_{L\cup R}$ (an `OR` condition). Using the same logic, the point χ will be within both L and R if its largest value comes back as negative, correctly capturing the intersection $F_{L\cap R}$ (an `AND` condition). For the difference F_{L-R}, F_R is negated since points outside R satisfy $F_R > 0$. So, choosing the largest value from $-F_R < 0$ and F_L grantees L.



(a) Union (b) Intersection (c) Difference

Figure 2: Boolean CSG operators to combine or cut parts of iso-surfaces to construct surface models.

In practice, before the boolean operations could be applied to a point, the value had to be converted from the world coordinate space to the local object space. The operator functions then worked with point and function values only, which is possible since the implicit surfaces are certain function values in an infinite object space.

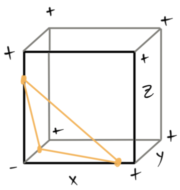


Figure 3: The marching cubes algorithm that decide sampling distance on iso-surfaces. Triangle(s) form as linear interpolation along the voxel edges is performed.

Secondly, in-, outside and on-surface points knowledge can be used to determine the sampling distance of the surfaces, in this case using the marching cubes algorithm. That way the smoothness of the surface can be altered. The marching cubes algorithm - refer to Figure 3 - subdivides the space into cubes - voxels - and measures the distance to the surface in those cubes that intersect with the surface. Voxel corners are marked with $-$ or $+$ if they lie in- or outside of the surface. By linearly interpolating along the voxel edges the zero-contour surface is defined as one or more triangles per voxel. Thus, decreasing the voxel size will produce shorter sampling distances, and thereby more and smaller triangles - i.e a smoother surface.

Function F can be specified to the general quadric form in Equation 2 and then be rewritten to matrix form in 3, for the zero-contour $C = 0$. The letters in the matrix - also called $\mathbf{Q}$ - designate coefficients of the quadric function. Thus, only by identifying and changing the values of these coefficients [A-J], different quadric surfaces can be derived - such as planes, paraboloids, cones and spheres to name a few. For example, a cylinder is given by Equation 4, which shows that all coefficients but those controlling the terms x^2 and y^2 can be set to zero ($B = C = D = F = G = H = I = 0, A = E \neq 0, J = -1$).

$$F(x,y,z) = Ax^2 + 2Bxy + 2Cxz + 2Dx + Ey^2 + 2Fyz + 2Gy + Hz^2 + 2Iz + J \quad (2)$$

$$F(x,y,z) = 0 = \begin{bmatrix} x & y & z & 1 \end{bmatrix} \begin{bmatrix} A & B & C & D \\ B & E & F & G \\ C & F & H & I \\ D & G & I & J \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} \quad (3)$$

$$1 = \frac{x^2}{a^2} + \frac{y^2}{a^2} \quad (4)$$

Another property of iso-surfaces is that their normals become precisely correct - in contrast the approximated normals of triangle meshes. This is because implicit surfaces are defined as - to some degree - continuous functions $F(x,y,z)$, and so the direction and magnitude in which the function grows/shrinks can be found from the gradient of F. By taking the partial derivatives for function F in Equation 3 the gradient ∇F results as Equation 5.

To implement Equation 5 in practice, the point coordinates firstly had to be converted to local object space, since that is the space in which the quadric surfaces are defined.

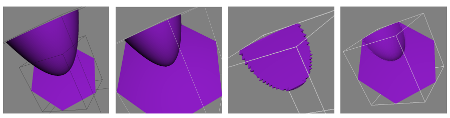
$$\nabla F = 2\begin{bmatrix} A & B & C & D \\ B & E & F & H \\ C & F & H & I \end{bmatrix}\begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} = 2\mathbf{Q}_{sub}\mathbf{p} \tag{5}$$

In addition to gradients, the normals of an implicit surface can be directly and accurately obtained from the gradient vector at any point on the surface. Since the gradient ∇F points in the direction of the greatest increase of the function $F(x, y, z)$, it is naturally perpendicular to the surface F itself. Therefore, the surface normal at a point is simply the normalized gradient, as is shown in Equation 6.

$$\bar{\mathbf{n}} = \pm\frac{\nabla F}{||\nabla F||} \tag{6}$$

2 Results

The boolean operators applied to a plane (D=G=I=1) and an elliptic paraboloid (A=1, E=1, I=$-\frac{1}{2}$) are shown in Figures 4(b) to 4(d). Figure 4 shows that the operations are correct in that they successfully merge the paraboloid and the plane in 4(b). For the intersection in 4(c) only that part of the plane and the paraboloid that touch in 4(a) are preserved. Finally, taking the difference plane-paraboloid removes from the plane the part of the paraboloid intersecting with it, leaving an empty mark on the plane in 4(d).

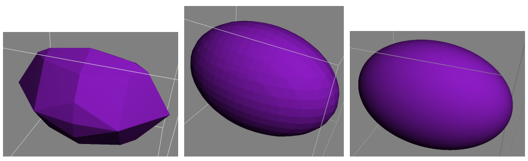


(a) Original surface combination (b) Union Result (c) Intersection Result (d) Difference Result

Figure 4: Results of applying the boolean operators in Equation 1 on two iso-surfaces elliptic paraboloid (A=1, E=1, I=$-\frac{1}{2}$) and a plane (D=G=I=1).

The results in Figure 4 is a testimony to how easy and relatively fast it is to model and reshape surfaces into desired constructions and combinations, through basic true or false logic.

Figure 5 furthermore exemplifies how the sampling distance affects the smoothness or resolution of the surface. The larger the voxels are (0.1 in Figure 5(a)), the coarser (fewer triangles) the iso-surface becomes. Conversely, smaller voxels (0.02 in 5(c)) smoothens the surface significantly, as more voxel edges to interpolate on results in more triangles. Since a function F defining an iso-surface is continuous, it would in theory be possible to achieve infinite resolution by simply letting the voxel size go to infinitely decreasing size. This is however not possible in practice as all machines have a computational limitation, and tasks have time limitations. Anyhow, these results still indicate that a sufficiently smooth surface can be obtained and that jaggedness produced in combined surfaces - like the intersection in Figure 4(c) - can be remedied by a resampling after the boolean operation.



(a) Sampling distance: 0.5 (b) Sampling distance: 0.1 (c) Sampling distance: 0.02

Figure 5: An implicit ellipsoid (A=1, E=2, H=3, J=-1) sampled with the distances 0.5, 0.1 and 0.02.

Finally, the accurateness of surface normals and gradients is showcased in Figure 6 for a hyperbolic paraboloid (A=1, E=-1, I=-$-\frac{1}{2}$). Visibly, lengths the gradients (blue) follow but direction and amount of slope along the surface. For example at the minimum point - or the saddle point - where the slope approaches zero, the gradient vectors become very short, in comparison to steeper sections like the top ridges furthest away from the saddle point.

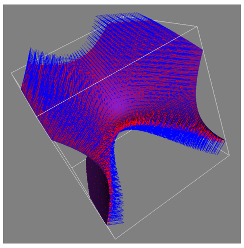


Figure 6: An implicit hyperbolic paraboloid (A=1, E=-1, I=-$-\frac{1}{2}$) with gradients in blue and surface normals in red.

For the normals, these keep exact orientation - unlike they do for triangle meshes where triangle contribution vary. Therefore, the derivatives of the function F describing implicit surfaces used to compute normals and gradients also work well to find the curvature κ.