CSE528 Computer Graphics: Theory, Algorithms, and Applications

Hong Qin
Department of Computer Science
State University of New York at Stony Brook (Stony Brook University)
Stony Brook, New York 11794--4400
Tel: (631)632-8450; Fax: (631)632-8334
qin@cs.sunysb.edu
http://www.cs.sunysb.edu/~qin
Implicit Surfaces
Straight Line (Implicit Representation)

\[ x + 2y - 4 = 0 \]

\[ x + 2y - 4 > 0 \]

\[ x + 2y - 4 < 0 \]
Straight Line

- Mathematics (Implicit Representation)

\[ ax + by + c = 0 \]
\[ + \alpha(ax + by + c) = 0 \]
\[ -\alpha(ax + y + c) = 0 \]

- Example

\[ x + 2y - 4 = 0 \]
Circle

- Implicit representation

\[ x^2 + y^2 - 1 > 0 \]

\[ x^2 + y^2 - 1 < 0 \]

\[ x^2 + y^2 - 1 = 0 \]
Conic Sections

- **Mathematics**
  \[ ax^2 + 2bxy + cy^2 + dx + ey + f = 0 \]

- **Examples**
  - Ellipse
    \[ 2x^2 + 3y^2 - 5 = 0 \]
  - Hyperbola
    \[ 2x^2 - 3y^2 - 5 = 0 \]
  - Parabola
    \[ 2x^2 + 3y = 0 \]
  - Empty set
    \[ 2x^2 + 3y^2 + 1 = 0 \]
  - Point
    \[ 2x^2 + 3y^2 = 0 \]
  - Pair of lines
    \[ 2x^2 - 3y^2 = 0 \]
  - Parallel lines
    \[ 2x^2 - 7 = 0 \]
  - Repeated lines
    \[ 2x^2 = 0 \]
Conics

- Parametric equations of conics
- Generalization to higher-degree curves
- How about non-planar (spatial) curves
Conics

\[ Ax^2 + 2Bxy + Cy^2 + 2Dx + 2Ey + F = 0 \]

\[ \mathbf{PQP}^T = 0 \]

\[ \mathbf{Q} = \begin{bmatrix} A & B & D \\ B & C & E \\ D & E & F \end{bmatrix} \]

\[ \mathbf{P} = [x \ y \ 1] \]

Table 2.1 Conic curve characteristics

| \( k \) | \( |\mathbf{Q}| \) | Other conditions | Type                  |
|--------|----------------|-----------------|---------------------|
| 0      | ≠ 0           |                  | Parabola            |
| 0      | 0             | \( C \neq 0, E^2 - CF > 0 \) | Two parallel real lines |
| 0      | 0             | \( C \neq 0, E^2 - CF = 0 \) | Two parallel coincident lines |
| 0      | 0             | \( C \neq 0, E^2 - CF < 0 \) | Two parallel imaginary lines |
| 0      | 0             | \( C = B = 0, D^2 - AF > 0 \) | Two parallel real lines |
| 0      | 0             | \( C = B = 0, D^2 - AF = 0 \) | Two parallel coincident lines |
| 0      | 0             | \( C = B = 0, D^2 - AF < 0 \) | Two parallel imaginary lines |
| <0     | 0             |                  | Point ellipse       |
| <0     | ≠ 0           | \(-C|\mathbf{Q}| > 0\) | Real ellipse        |
| <0     | ≠ 0           | \(-C|\mathbf{Q}| < 0\) | Imaginary ellipse   |
| <0     | ≠ 0           |                  | Hyperbola           |
| <0     | 0             |                  | Two intersecting lines |
Plane

\[ x + y + z - 1 = 0 \]
Plane and Intersection

\[ p_a, p_c, p_b \]

\[ \mathbf{n} \]
Plane

- **Example**  
  \[ x + y + z - 1 = 0 \]

- **General plane equation**  
  \[ ax + by + cz + y = 0 \]

- **Normal of the plane**  
  \[ \mathbf{n} = \begin{bmatrix} a \\ b \\ c \end{bmatrix} \]

- **Arbitrary point on the plane**  
  \[ \mathbf{p}_a = \begin{bmatrix} a_x \\ a_y \\ a_z \end{bmatrix} \]
Plane

• Plane equation derivation

\[(x - a_x)a + (y - a_y)b + (z - a_z)c = 0\]
\[ax + by + cz - (a_xa + a_yb + a_zc) = 0\]

• Parametric representation (given three points on the plane and they are non-collinear!)

\[p(u, v) = p_a + (p_b - p_a)u + (p_c - p_a)v\]
Plane

- **Explicit expression (if c is non-zero)**
  \[ z = -\frac{1}{c} (ax + by + d) \]

- **Line-Plane intersection**
  \[
  l(u) = p_0 + (p_1 - p_0)u \\
  (n)(p_0 + (p_1 - p_0)u) + d = 0 \\
  u = -\frac{np_0}{np_1 - np_0} = -\frac{plane(p_0)}{plane(p_1) - plane(p_0)}
  \]
Circle

- Implicit equation: \( x^2 + y^2 - 1 = 0 \)

- Parametric function: 
  \[
  c(\theta) = \begin{bmatrix}
  \cos(\theta) \\
  \sin(\theta)
  \end{bmatrix}
  \\ 0 \leq \theta \leq 2\pi
  \]

- Parametric representation using rational polynomials (the first quadrant): 
  \[
  x(u) = \frac{1 - u^2}{1 + u^2} \\
  y(u) = \frac{2u}{1 + u^2} \\
  u \in [0,1]
  \]

- Parametric representation is not unique!
What are Implicit Surfaces?

- 2D Geometric shapes that exist in 3D space
- Surface representation through a function $f(x, y, z) = 0$
- Most methods of analysis assume $f$ is continuous and not everywhere 0.
Example of an Implicit Surface

• 3D Sphere centered at the origin

\[ x^2 + y^2 + z^2 = r^2 \]

\[ x^2 + y^2 + z^2 - r^2 = 0 \]
Point Classification

- **Inside Region**: $f < 0$
- **Outside Region**: $f > 0$
- Or vice versa depending on the function

$$f = 0$$

$$f < 0$$

$$f > 0$$
Manifold

- A 2D Manifold separates space into a natural inner and natural outer region
- A manifold surface contains no holes or dangling edges
Manifold

- It is difficult to determine enclosed region in non-manifold surfaces
Surface Normals

• Usually gradient of the function
  \[ \nabla f(x,y,z) = \left( \frac{\partial f}{\partial x}, \frac{\partial f}{\partial y}, \frac{\partial f}{\partial z} \right) \]

• Points at increasing f
Properties of Implicits

• Easy to check if a point is inside the implicit surface or NOT
  – Simply evaluate \( f \) at that point

• Fairly easy to check ray intersection
  – Substitute ray equation into \( f \) for simple functions
  – Binary search
$z = f(x, y)$

$f(x, y) = 0$
Implicit Equations for Curves

• Describe an implicit relationship

• Planar curve (point set) \( \{(x, y) \mid f(x, y) = 0\} \)

• The implicit function is not unique

\[
\{(x, y) \mid +\alpha f(x, y) = 0\}
\]

\[
\{(x, y) \mid -\alpha f(x, y) = 0\}
\]

• Comparison with parametric representation

\[
p(u) = \begin{bmatrix} x(u) \\ y(u) \end{bmatrix}
\]
Implicit Equations for Curves

- **Implicit function is a level-set**
  \[
  \begin{cases}
  z = f(x, y) \\
  z = 0
  \end{cases}
  \]

- **Examples (straight line and conic sections)**
  \[
  \begin{align*}
  ax + by + c &= 0 \\
  ax^2 + 2bxy + cy^2 + dx + ey + f &= 0
  \end{align*}
  \]

- **Other examples**
  - Parabola, two parallel lines, ellipse, hyperbola, two intersection lines
Implicit Functions for Curves

- Parametric equations of conics
- Generalization to higher-degree curves
- How about non-planar (spatial) curves
Implicit Equations for Surfaces

- **Surface mathematics** \( \{(x, y, z) \mid f(x, y, z) = 0\} \)
- **Again, the implicit function for surfaces is not unique**
  \( \{(x, y, z) \mid +\alpha f(x, y, z) = 0\} \), \( \{(x, y, z) \mid -\alpha f(x, y, z) = 0\} \)
- **Comparison with parametric representation**

\[ p(u, v) = \begin{bmatrix} x(u, v) \\ y(u, v) \\ z(u, v) \end{bmatrix} \]
Implicit Equations for Surfaces

- **Surface defined by implicit function is a level-set**
  \[
  \begin{align*}
  w &= f(x, y, z) \\
  w &= 0
  \end{align*}
  \]

- **Examples**
  - Plane, quadric surfaces, tori, superquadrics, blobby objects

- **Parametric representation of quadric surfaces**

- **Generalization to higher-degree surfaces**
Quadric Surfaces

- **Implicit functions**

- **Examples**
  - Sphere
  - Cylinder
  - Cone
  - Paraboloid
  - Ellipsoid
  - Hyperboloid

- **More**
  - Two parallel planes, two intersecting planes, single plane, line, point

\[ ax^2 + by^2 + cz^2 + dxy + exz + fyz + gx + hy + jz + k = 0 \]

\[
\begin{align*}
  x^2 + y^2 + z^2 - 1 &= 0 \\
  x^2 + y^2 - 1 &= 0 \\
  x^2 + y^2 - z^2 &= 0 \\
  x^2 + y^2 + z &= 0 \\
  2x^2 + 3y^2 + 4z^2 - 5 &= 0 \\
  x^2 + y^2 - z^2 + 4 &= 0
\end{align*}
\]
Quadric Surfaces

- **Implicit surface equation**

\[ f(x, y, z) = ax^2 + by^2 + cz^2 + 2dxy + 2eyz + 2fyz + 2gx + 2hy + 2jz + k = 0 \]

- **An alternative representation**

\[
P^T \cdot Q \cdot P = 0
\]

with

\[
Q = \begin{bmatrix}
a & d & f & g \\
d & b & e & h \\
f & e & c & j \\
g & h & j & k \\
\end{bmatrix}
\]

\[
P = \begin{bmatrix}
x \\
y \\
z \\
1 \\
\end{bmatrix}
\]

- **Sphere**
  
  \[ x^2 + y^2 + z^2 - r^2 = 0 \]
  
  \[ x = r \cos(\alpha) \cos(\beta) \]
  
  \[ y = r \cos(\alpha) \sin(\beta) \]
  
  \[ z = r \sin(\alpha) \]
  
  \[ \alpha \in \left[-\frac{\pi}{2}, \frac{\pi}{2}\right]; \beta \in [-\pi, \pi] \]

- **Ellipsoid**
  
  \[ \frac{x^2}{a^2} + \frac{y^2}{b^2} + \frac{z^2}{c^2} - 1 = 0 \]
  
  \[ x = a \cos(\alpha) \cos(\beta) \]
  
  \[ y = b \cos(\alpha) \sin(\beta) \]
  
  \[ z = c \sin(\alpha) \]
  
  \[ \alpha \in \left[-\frac{\pi}{2}, \frac{\pi}{2}\right]; \beta \in [-\pi, \pi] \]

- **Geometric meaning of these parameters**
Quadric Surfaces

- Modeling advantages
  - computing the surface normal
  - testing whether a point is on the surface
  - computing $z$ given $x$ and $y$
  - calculating intersections of one surface with another
Superquadrics

- **Geometry** (generalization of quadrics)
- **Superellipse**
- **Superellipsoid**

- **Parametric representation**

\[
\left(\frac{x}{a_1}\right)^{\frac{2}{s}} + \left(\frac{y}{a_2}\right)^{\frac{2}{s}} - 1 = 0
\]

\[
\left(\frac{x}{a_1}\right)^{\frac{2}{s_2}} + \left(\frac{y}{a_2}\right)^{\frac{2}{s_2}} + \left(\frac{z}{a_3}\right)^{\frac{s_2}{s_1}} - 1 = 0
\]

\[
\begin{bmatrix}
  x \\
  y \\
  z
\end{bmatrix} =
\begin{bmatrix}
  a_1 \cos^{s_1}(\alpha) \sin^{s_2}(\beta) \\
  a_2 \cos^{s_1}(\alpha) \sin^{s_2}(\beta) \\
  a_3 \sin^{s_2}(\alpha)
\end{bmatrix}
\]

\[
\alpha \in \left[-\frac{\pi}{2}, \frac{\pi}{2}\right]; \beta \in \left[-\pi, \pi\right)
\]

- **What is the meaning of these control parameters?**
Types of Implicit Surfaces

• **Mathematic**
  – Polynomial or *Algebraic*
  – Non polynomial or *Transcendental*
    • Exponential, trigonometric, etc.

• **Procedural**
  – Black box function
Generalization

- Higher-degree polynomials

\[ \sum \sum \sum a_{ijk} x^i y^j z^k = 0 \]

- Non polynomials
Algebraic Function

- Parametric representation is popular, but…
- Formulation
  \[ \sum \sum \sum a_{ijk} x^i y^j z^k = 0 \]
- Properties…
  - Powerful, but lack of modeling tools
Algebraic Surfaces

Cubic

Degree 4

Degree 6
Non-Algebraic Surfaces
Algebraic Patch

Tetrahedron

Control point, weight

Algebraic patch
Algebraic Patch

- A tetrahedron with non-planar vertices

$$V_{n000}, V_{0n00}, V_{00n0}, V_{000n}$$

- Trivariate barycentric coordinate \((r,s,t,u)\) for \(p\)

$$p = rV_{n000} + sV_{0n00} + tV_{00n0} + uV_{000n}$$

$$r + s + t + u = 1$$

- A regular lattice of control points and weights

$$p_{ijkl} = \frac{iv_{n000} + jv_{0n00} + kv_{00n0} + lv_{000n}}{n}$$

\(i, j, k, l \geq 0; i + j + k + l = n\)
Algebraic Patch

• There are \((n+1)(n+2)(n+3)/6\) control points. A weight \(w(I,j,k,l)\) is also assigned to each control point

• Algebraic patch formulation

\[
\sum \sum \sum \sum w_{ijkl} \frac{n!}{i! j! k! l!} r^i s^j t^k u^l = 0
\]

• Properties

  – Meaningful control, local control, boundary interpolation, gradient control, self-intersection avoidance, continuity condition across the boundaries, subdivision
Spatial Curves

• Intersection of two surfaces

\[
\begin{align*}
  f(x, y, z) &= 0 \\
  g(x, y, z) &= 0
\end{align*}
\]
Algebraic Solid

- **Half space**
  \[(x, y, z) | f(x, y, z) \leq 0\]; or
  \[(x, y, z) | f(x, y, z) \geq 0\]

- **Useful for complex objects (refer to notes on solid modeling)**

\[f(x, y, z) = \begin{bmatrix} f_1(x, y, z) \\ f_2(x, y, z) \\ f_3(x, y, z) \\ \vdots \end{bmatrix} = 0\]
Implicit Surfaces: Applications

- Zero sets of implicit functions.
  
  \[ f(x, y, z) = 0 \]
  
  \[ r^2 - x^2 - y^2 - z^2 > 0 \]

- CSG operations.
  
  \[ (l - |x| > 0) \cap (l - |y| > 0) \cap (l - |z| > 0) \]
Radial Basis Function: Applications


\[ f(x) = \sum \lambda_i \Phi(x - c_i) + p(x) \]

RBF fitting

Visualization of \( f=0 \)
Implicit Functions

- Long history: classical algebraic geometry
- Implicit and parametric forms
  - Advantages
  - Disadvantages
- Curves, surfaces, solids in higher-dimension
- Intersection computation
- Point classification
- Larger than parameter-based modeling
- Unbounded geometry
- Object traversal
- Evaluation
Implicit Functions

- Efficient algorithms, toolkits, software
- Computer-based shape modeling and design
- Geometric degeneracy and anomaly
- Algebraic and geometric operations are often closed
- Mathematics: algebraic geometry
- Symbolic computation
- Deformation and transformation
- Shape editing, rendering, and control
Implicit Functions

• Conversion between parametric and implicit forms
• Implicitization vs. parameterization
• Strategy: integration of both techniques
• Approximation using parametric models
Polygonization

- Conversion of implicit surface to polygonal mesh
- Display implicit surface using polygons
- Real-time approximate visualization method
- Two steps
  - Partition space into cells
  - Fit a polygon to surface in each cell
Polygonal Representation

- Partition space into convex cells
- Find cells that intersect the surface
  (traverse cells)
- Compute surface vertices
Cell Polygonization

- We will need to find those cells that actually contain parts of surface
- Need to approximate surface within cell
- Basic idea: use piecewise-linear approximation (polygon)
Implicit Surface (Polygonal Representation)

\[ F : \mathbb{R}^3 \rightarrow \mathbb{R}, \Sigma = F^{-1}(0) \]
Spatial Partitioning

- **Exhaustive enumeration**
  - Divide space into regular lattice of cells
  - Traverse cells in order to arrive at polygonization
Space Partitioning Criteria

How do we know if a cell actually contains the surface?

• Straddling Cells
  – At least one vertex inside and outside surface
  – Non-straddling cells can still contain surface

• Guarantees
  – Interval analysis
  – Lipschitz condition
Spatial Partitioning

- **Subdivision**
  - Start with root cell and subdivide
  - Continue subdividing
  - traverse cells
Spatial Partitioning

- Adaptive polygonization
Surface Vertex Computations

- Determine where implicit surface intersects cell edges
- EITHER linear interpolate function values to approximate
- OR numerically find zero of \( f(\mathbf{r}(t)) \)

\[
\mathbf{r}(t) = \mathbf{x}_1 + t(\mathbf{x}_2 - \mathbf{x}_1)
\]

\[
0 \leq t \leq 1
\]

\[
\mathbf{x} = \frac{\nu_1}{\nu_1 + \nu_2} \mathbf{x}_1 + \frac{\nu_2}{\nu_1 + \nu_2} \mathbf{x}_2
\]

\[
f(\mathbf{x}_1) = \nu_1 (+)
\]

\[
f(\mathbf{x}_2) = \nu_2 (-)
\]
Polygonal Shape

- Use table indexed by vertex signs and consider all possible combinations.
- Let + be 1, - be 0.
- Table size:
  - Tetrahedral cells: 16 entries.
  - Cubic cells: 256 entries.
- E.g., 2-D - 16 square cells.
Determining Intersections
Tetrahedral Cell Polygons
Orientation

- Consistency allows polygons to be drawn with correct orientation
- Supports backface culling
Problem: Ambiguity

- Some cell-corner-value configurations yield more than one consistent polygon
- Only for cubes, not tetrahedra (why?)
- In 3-D can yield holes in surface!
- How can we resolve these ambiguities?
Topological Inference

- Sample a point in the center of the ambiguous face
- If data is discretely sampled, bilinearly interpolate samples

\[ p(s,t) = (1-s)(1-t) \ a + s \ (1-t) \ b + (1-s) \ t \ c + s \ t \ d \]
Preferred Polarity

- Assume ambiguous face centers always +
- (or always –)
- Preference can be encoded into table
CSG Polygonization

• Polygonization can smooth crease edges caused by CSG operations
• Polygonization needs to add polygon vertices along crease edges
Visualization of Implicit Surfaces

Ray-tracing

Polygonization
(e.g. Marching cubes method)
Problem of Polygonization

- Sharp features are broken

50^3 grid

100^3 grid

200^3 grid
Reconstruction of Sharp Features

Input

Implicit function: \( f(x, y, z) \)

and

Rough Polygonization (Correct topology)

Post-processing

Output
Implicit Surfaces vs Polygons

• **Advantages**
  – Smoother and more precise
  – More compact
  – Easier to interpolate and deform

• **Disadvantages**
  – More difficult to display in real time
Implicits vs Parameter-Based Representations

• Advantages
  – Implicits are easier to blend and morph
  – Interior/Exterior description
  – Ray-trace

• Disadvantages
  – Rendering
  – Control
Case Studies: Distance Functions

- $D(p) = R$
  - Sphere: Distance to a point
  - Cylinder: Distance to a line
  - More examples
Distance Functions
Blobby Models

- Blobby models [Blinn 82], also known as metaballs [Nishimura and Hirai 85] or soft objects [Wyvill and Wyvill 86, 88]

- A blobby model — a center surrounded by a density field, where the density attributed to the center decreases with distance from the center.

- By simply summing the influences of each blobby model on a given location, we can obtain very smooth blends of the spherical density fields.

\[ G(x, y, z) = \sum_{i} g_i(x, y, z) - \text{threshold} = 0 \]
Blobs and Metaballs

- Define the location of some points
- For each point, define a function on the distance to a given point, \((x, y, z)\)
- Sum these functions up, and use them as an implicit function

Question: If I have two special points, in 2D, and my function is just the distance, what shape results?

More generally, use Gaussian functions of distance, or other forms

- Various results are called blobs or metaballs
Design Using Blobs

• None of these parameters allow the designer to specify exactly where the surface is actually located.
• A designer only has indirect control over the shape of a blobby implicit surface.
• Blobby models facilitate the design of smooth, complex, organic-appearing shapes.
Example with Blobs

Rendered with POVray. Not everything is a blob, but the characters are.
What Is It?

- “Metaball, or ‘Blobby’, Modeling is a technique which uses implicit surfaces to produce models which seem more ‘organic’ or ‘blobby’ than conventional models built from flat planes and rigid angles”
Examples
Examples
Blobby Modeling: Its Utility

- Organic forms and nonlinear shapes
- Scientific modeling (electron orbitals, some medical imaging)
- Muscles and joints with skin
- Rapid prototyping
- CAD/CAM solid geometry
Blobby Model and Mathematics

• Implicit equation:

\[ f(x, y, z) = \sum_{i=1}^{n_{blobs}} w_i g_i(x, y, z) = d \]

• The \( w_i \) are weights – just numbers

• The \( g_i \) are functions, one common choice is:

\[ g_i(x) = e^{\frac{-(x-c_i)^2}{\sigma_i^2}} \]

– \( c_i \) and \( \sigma_i \) are parameters
Skeletal Design

- Use skeleton technique to design implicit surfaces and solids toward interactive speed.
- Each skeletal element is associated with a locally defined implicit function.
- These local functions are blended using a polynomial weighting function.
  - [Bloomenthal and Wyvill 90, 95, 97] defined skeletons consisting of points, splines, polygons.
  - 3D skeletons [Witkin and Heckbert 94] [Chen 01]
Skeletal Design

- **Global and local control in three separate ways:**
  - Defining or manipulating of the skeleton;
  - Defining or adjusting those implicit functions defined for each skeletal element;
  - Defining a blending function to weight the individual implicit functions.
Rendering Implicit Surfaces

• Some methods can render then directly
  – Raytracing - find intersections with Newton’s method

• For polygonal renderer, must convert to polygons

• Advantages:
  – Good for organic looking shapes e.g., human body
  – Reasonable interfaces for design

• Disadvantages:
  – Difficult to render and control when animating
  – Being replaced with subdivision surfaces, it appears
Display Implicit Surfaces

- **Recursive subdivision:**

![Diagram of recursive subdivision of an implicit surface]
Display Implicit Surfaces

- **Recursive subdivision:**
Display Implicit Surfaces

- Recursive subdivision:
Display Implicit Surfaces

• Find the edges, separating hot from cold:
Compression

Mesh of 473,000 vertices and 871,000 facets

Implicit function of 32,000 terms
Deformation

- $p' = D(p)$
- $D$ maps each point in 3-space to some new location
- Twist, bend, taper, and offset
Visualization

- Contours
Visualization

• Particle Display
Particle Systems

- Witkin Heckbert S94
- Constrain particle system to implicit surface
  (Implicit surface $f = 0$ becomes constraint surface $C = 0$)
- Particles exert repulsion forces onto each other to spread out across surface
- Particles subdivide to fill open gaps
- Particles commit suicide if overcrowded
- Display particle as oriented disk
  - Constrain implicit surface to particles!
Meshing Particles

- **Stander Hart S97**
- **Use particles as vertices**
- **Connect vertices into mesh**
- **Problems:**
  - Which vertices should be connected?
  - How should vertices be reconnected when surface moves?
- **Solution:** Morse theory
- **Track/find critical points of function in topology of implicit surface**
Shrink-wrapping Mechanism

- Look at family of surfaces $f^{-1}(s)$ for $s > 0$
- For $s$ large, $f^{-1}(s)$ spherical
- Polygonize sphere
- Reduce $s$ to zero
  - Allow vertices to track surface
  - Subdivide polygons as necessary when curvature increases
Visualization

• Ray Tracing
Other Coordinate Systems

Spherical Coordinates

Cylindrical Coordinates
Summary

• Surface defined implicitly by \( f(p) = 0 \)
• Easy to test if point is on surface, inside, or outside
• Easy to handle blending, interpolation, and deformation
• Difficult to render