3.2 Surfaces

Reading

Recommended:

- Foley, et al., Chapter 11
- Bartels, Beatty, and Barsky, *An Introduction to Splines for use in Computer Graphics and Geometric Modeling*, 1987.
- Stollnitz, DeRose, and Salesin, *Wavelets for Computer Graphics: Theory and Applications*, 1996. Sections 10.2.

Outline

1. Surfaces of revolution
2. General sweep surfaces
3. Tensor product surfaces
4. Subdivision surfaces

Surfaces of revolution

Idea: Rotate a 2D "profile" curve around an axis.

Q: What kinds of shapes can you model this way?

Constructing a surface of revolution

Given: A curve $C(u) : u \in [0,1]$ in the yz-plane:

$$C(u) = \begin{bmatrix} 0 \\ c_y(u) \\ c_z(u) \\ 1 \end{bmatrix}$$

Let: $R_z(\theta)$ be a rotation matrix about the z-axis.

Find: A surface $S(u,v) : u,v \in [0,1]$, which is $C(u)$ rotated about the z-axis.

Solution:

General sweep surfaces

The "surface of revolution" is a special case of a "sweep surface"....

Idea: Trace out surface $S(u,v)$ by moving a "profile" curve $C(u)$ along a "trajectory" $T(v)$.

More specifically:

- Suppose that $C(u)$ lies in an $(\bar{x}_c, \bar{y}_c)$ coordinate system with origin $\mathcal{O}_c$.
- For every point along $T(v)$, lay in $C(u)$ so that $T(v)$ coincides with $\mathcal{O}_c$.

Orientation

The big issue:

- How to orient $C(u)$ as it moves along $T(v)$.

Lots of options. Here are two:

1. Fixed (or "static"): Just translate $\mathcal{O}_c$ along $T(v)$:

2. Moving: Use the "Frenet frame" of $T(v)$.
   - · Allows smoothly varying orientation
   - · Permits surfaces of revolution, for example

Frenet frames

Motivation: Given a curve $T(v)$, we want to attach a smoothly-varying coordinate system.

Idea: To get a 3D coordinate system, we need 3 independent direction vectors.

- Tangent $\hat{t}(v) = \mathit{normalize}(T'(v))$
- Bi-normal $\hat{b}(v) = \mathit{normalize}(T'(v) \times T''(v))$
- Normal $\hat{n}(v) = \hat{b}(v) \times \hat{t}(v)$

As we move along $T(v)$, the Frenet frame $(\hat{t}, \hat{b}, \hat{n})$ varies smoothly.

Frenet swept surfaces

Idea: Orient the profile curve $C(u)$ using the Frenet frame of the trajectory $T(v)$:

- Put $C(u)$ in the "normal" plane $\hat{n}\hat{b}$.
- Place $\mathcal{O}_c$ on $T(v)$.
- Align $\hat{x}_c$ for $C(u)$ with $-\hat{n}$.
- Align $\hat{y}_c$ for $C(u)$ with $\hat{b}$.

Cool: If $T(v)$ is a circle, you get a surface of revolution exactly!

Problem: What do you do when the curvature changes sign or goes to 0?

Variations

Several variations are possible:

- Scale $C(u)$ as it moves, possibly using length of $T(v)$ as a scale factor — great for seashells!
- Morph $C(u)$ into some other curve $C'(u)$ as it moves along $T(v)$.
- Use your imagination....

Bézier tensor product surfaces

Given: A grid of control points $V_{ij} : i, j = 0, \ldots, n$.

Construct: A surface $S(u, v)$ by

- Treating rows of V as control points for curves $C_0(u), \ldots, C_n(u)$.
- Treating $C_0(u), \ldots, C_n(u)$ as control points for a curve parameterized by v.

Q: Which control points are interpolated by the surface?

Matrix form

Tensor product surfaces can be written out explicitly:

$$\begin{aligned} S(u,v) &= \sum_{i,j=0,\ldots,n} V_{ij} B_i^n(u) B_j^n(v) \\ &= \begin{bmatrix} v^3 & v^2 & v & 1 \end{bmatrix} M_{\text{Bézier}}\, V\, M_{\text{Bézier}}^{\text{T}} \begin{bmatrix} u^3 \\ u^2 \\ u \\ 1 \end{bmatrix} \end{aligned}$$

Subdivision surfaces

Chaikin's use of subdivision for curves inspired similar techniques for surfaces....

Idea: Iteratively refine a *control polyhedron* (or *control mesh*) $M^0 \to M^1 \to \cdots$ to a limit surface

$$\sigma = \lim_{j \to \infty} M^j$$

using splitting and averaging steps:

Splitting and averaging steps (Stollnitz et al., 10.4)

Two types of splitting steps:

- *Vertex schemes*
- *Face schemes*

Vertex schemes

A vertex surrounded by n faces is split into n subvertices, one for each face:

Doo-Sabin subdivision (Stollnitz et al., 10.3)

Face schemes

Each quadrilateral face is split into four subfaces:

Catmull-Clark subdivision (Stollnitz et al., 10.2)

Face schemes, cont.

Each triangular face is split into four subfaces:

Loop subdivision (Stollnitz et al., 10.5)

Averaging step

Once again we can use *masks* for the averaging step:

where

$$\alpha(n) = \frac{n\,(1-\beta(n))}{\beta(n)} \qquad \text{and} \qquad \beta(n) = \frac{5}{4} - \frac{(3+2\cos(2\pi/n))^2}{32}$$

(α carefully chosen to ensure smoothness.)

Interpolating subdivision surfaces

Interpolating schemes are defined by:

- Splitting
- Averaging only new vertices

Here is the averaging mask for the "butterfly scheme":

Smoothness increases as t goes from 0 to 1/8
(Stollnitz et al., 10.7)