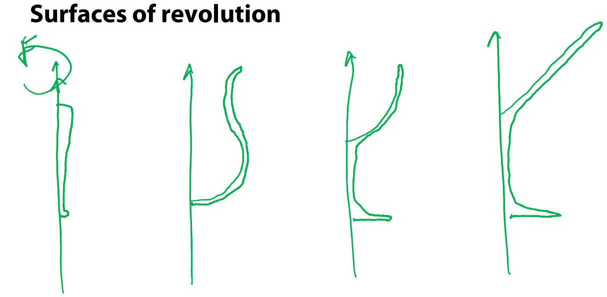
Surfaces of Revolution

Brian Curless
CSE 557
Autumn 2017

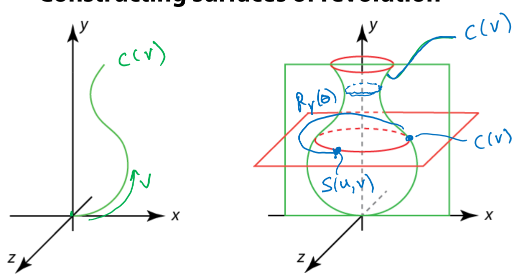
Surfaces of revolution



Idea: rotate a 2D **profile curve** around an axis.

What kinds of shapes can you model this way?

Constructing surfaces of revolution



Given: A curve $C(v)$ in the xy-plane:

$$C(v) = \begin{bmatrix} C_x(v) \\ C_y(v) \\ 0 \\ 1 \end{bmatrix}$$

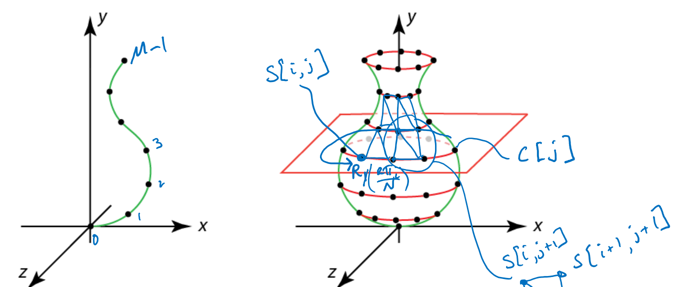
Let $R_y(\theta)$ be a rotation about the y-axis.

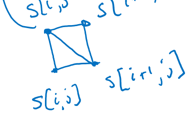
Find: A surface $S(u,v)$ which is $C(v)$ rotated about the y-axis, where $u,v \in [0, 1]$.

Solution: $R_y(2\pi u)\, C(v)$

Constructing surfaces of revolution

We can sample in u and v to get a grid of points over the surface.





Suppose we sample:

- in v, to give $C[j]$ where $j \in [0..M-1]$
- in u, to give rotation angle $\theta[i] = 2\pi i/N$ where $i \in [0..N]$

We can now write the surface as:

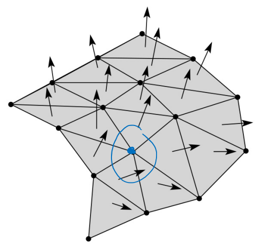
$$S[i,j] = R_y\left(\frac{2\pi i}{N}\right) C[j]$$

How would we turn this into a mesh of triangles?
How do we assign per-vertex normals?

Surface normals

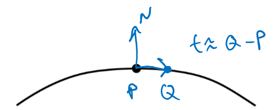
Now that we describe the surface as a triangle mesh, we need to provide surface normals. As we'll see later, these normals are important for drawing and shading the surface (i.e., for "rendering").

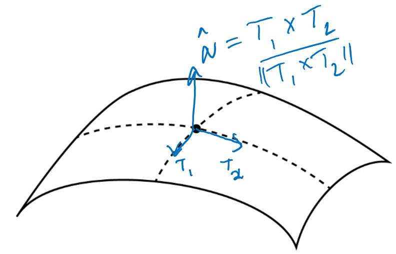
One approach is to compute the normal to each triangle. How do we compute these normals?



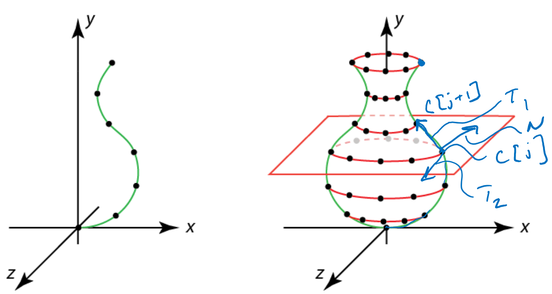
For surfaces of revolution, we can get better-looking results by analytically computing the normal at each vertex…

Tangent vectors, tangent planes, and normals





Normals on a surface of revolution



We can compute tangents to the curve points in the xy-plane:

$\mathbf{T}_1[0, j] \approx C[j+1] - C[j]$

$\mathbf{T}_2[0, j] = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$

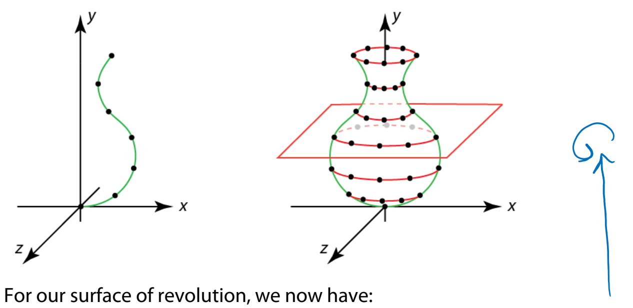
to get the normal in that plane:

$\mathbf{N}[0, j] = T_1[0,j] \times T_2[0,j]$

and then rotate it around:

$N[i,j] = R_y(2\pi i)\, N[0,j]$

Texture coordinates on a surface of revolution



For our surface of revolution, we now have:

Profile curve: $C[j]$ where $j \in [0..M-1]$

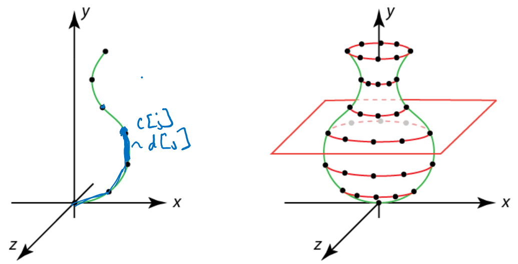
Rotation angles: $\theta[i] = 2\pi i / N$ where $i \in [0..N]$

The simplest assignment of texture coordinates would be:

$u = \frac{i}{N}$ $\quad v = \frac{j}{M-1}$

Note that you should include the rotation angles for $i = 0$ and $i = N$, even though they produce the same points (after rotating by 0 and 2π). Why do this??

Texture coordinates on a surface of revolution



We can get distortion in v if the samples are not evenly spaced along the profile curve.

We can reduce this distortion in v. Define:

$$d[j] = \begin{cases} \left\| C[j] - C[j-1] \right\|, & \text{if } j \neq 0 \\ 0, & \text{if } j = 0 \end{cases}$$

and set v to fractional distance along the curve:

$$v = \frac{\sum_{k=0}^{j} d[j]}{\sum_{k=0}^{M-1} d[j]}$$

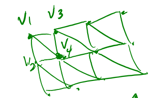
You must do this for v for the programming assignment!

Triangle meshes

How should we generally represent triangle meshes?

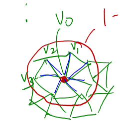
$V_1, N_1, u_1, v_1,$
V_2, N_2
V_4, N_4

V_1, N_1
V_4, N_4
V_3, N_3



Vertex list
V_1, N_1, u_1, v_1
V_2, N_2, u_2, v_2
V_3 :
V_4 :
V_5
:

$\triangle$ index list
1, 2, 4
1, 4, 3



1-ring neighborhood
N neighboring vertices
$\Rightarrow$ valence N

$$V_0' = \frac{V_0 + \frac{a}{N} V_1 + \frac{a}{N} V_2 + \cdots}{\underbrace{1 + \frac{a}{N} + \frac{a}{N} + \cdots}_{1+a}}$$

$\Rightarrow$ re-compute normals as avg. of face normals