CS-C3100 Computer Graphics

4.3 Procedural and Implicit Surfaces

Representing Surfaces

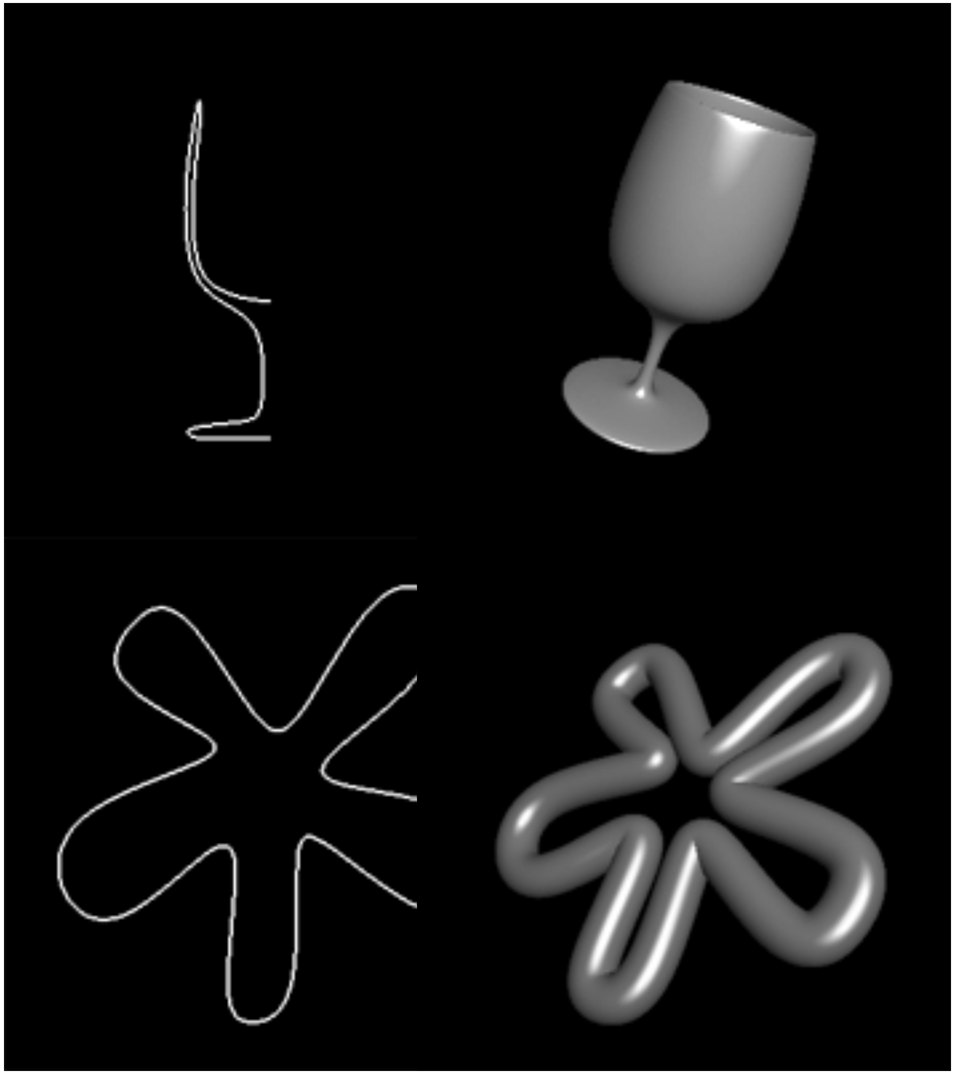
- Triangle meshes ✔
  - Surface analogue of polylines, this is what GPUs draw
- Tensor Product Splines
  - Surface analogue of spline curves
- Subdivision surfaces
- **Implicit surfaces**
  - $f(x,y,z)=0$
- **Procedural**
  - e.g. surfaces of revolution, generalized cylinder
- From volume data (medical images, etc.)

In These Slides

- Procedural surfaces
  - Swept surfaces
  - Surfaces of revolution
- Implicit surfaces

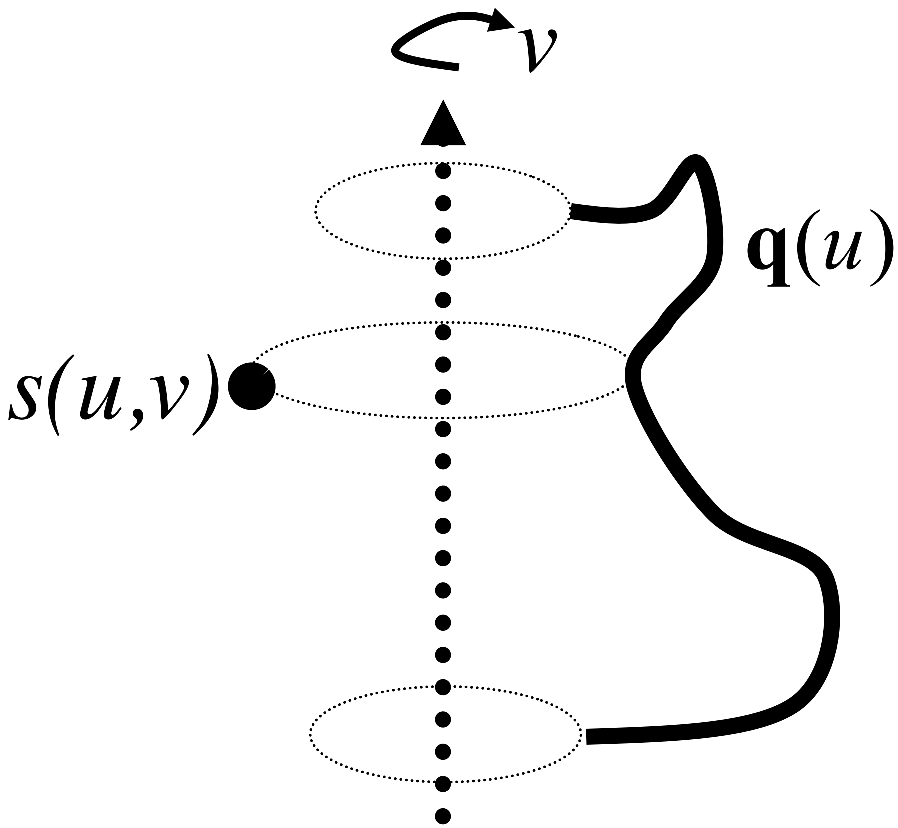
Specialized Procedural Definitions

- Surfaces of revolution
  - Rotate given 2D profile curve
- Generalized cylinders
  - Given 2D profile and 3D curve, sweep the profile along the 3D curve
- Assignment 2 extras!



Surface of Revolution

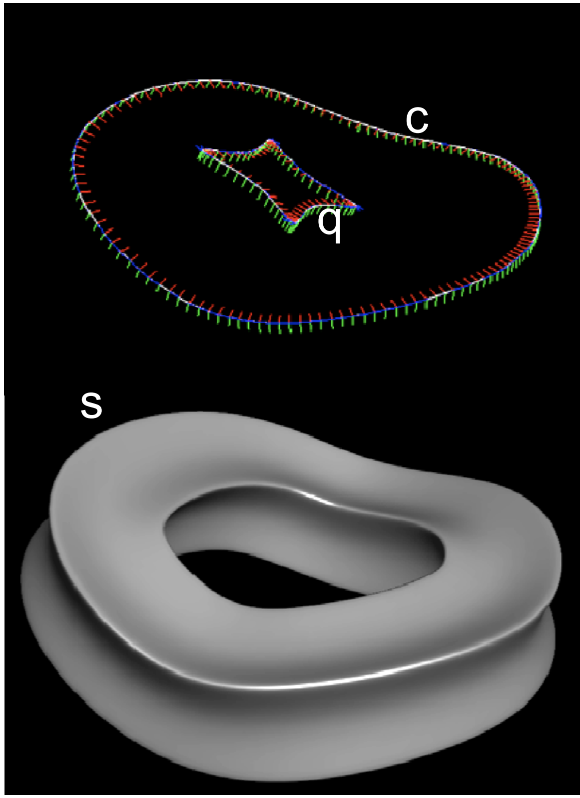
- 2D curve q(u) provides one dimension
  - Note: works also with 3D curve
- Rotation R(v) provides 2nd dimension



$\boldsymbol{s}(u,v)=\boldsymbol{R}(v)\boldsymbol{q}(u)$
where $\boldsymbol{R}$ is a matrix,
$\boldsymbol{q}$ a vector,
and $\boldsymbol{s}$ is a point on the surface

General Swept Surfaces

- Trace out surface by moving a profile curve along a trajectory.
  - profile curve $\mathbf{q}(u)$ provides one dim
  - trajectory $\mathbf{c}(u)$ provides the other
- Surface of revolution can be seen as a special case where trajectory is a circle

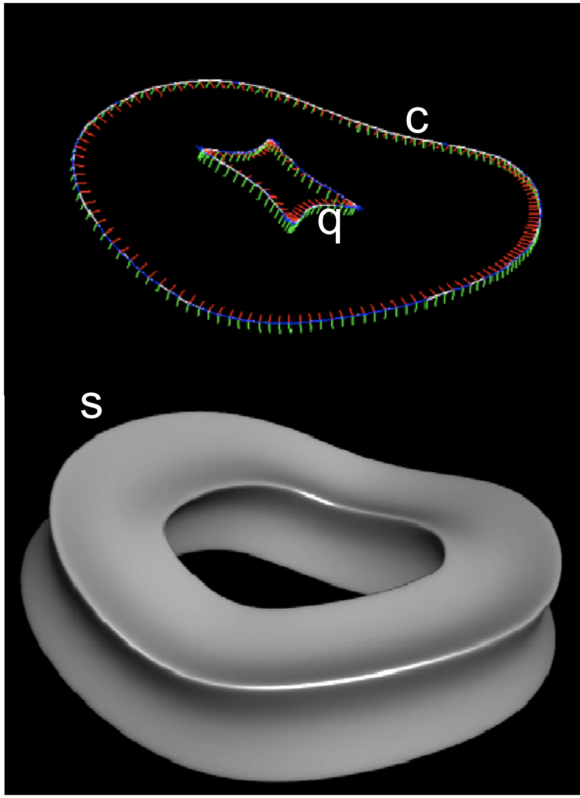


$$\boldsymbol{s}(u,v)=\boldsymbol{M}(\boldsymbol{c}(v))\boldsymbol{q}(u)$$

where $\boldsymbol{M}$ is a matrix that depends on the trajectory $\boldsymbol{c}$

General Swept Surfaces

- How do we get **M**?
  - Translation is easy, given by ***c****(v)*
  - What about orientation?
- Orientation options:
  - Align profile curve with an axis.
  - **Better**: Align profile curve with frame that "follows" the curve



$$\boldsymbol{s}(u,v)=\boldsymbol{M}(\boldsymbol{c}(v))\boldsymbol{q}(u)$$

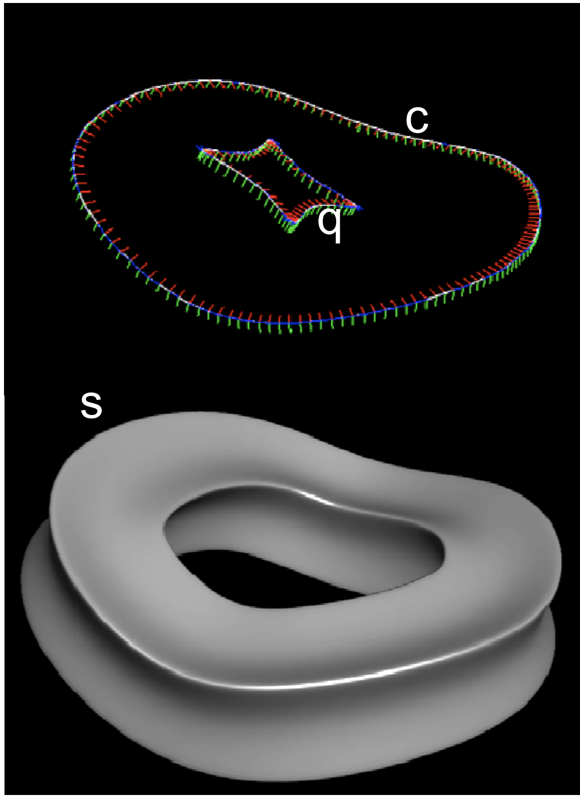
where ***M*** is a matrix that depends on the trajectory ***c***

Normals for Swept Surfaces

- Need partial derivatives w.r.t. both u and v

$$n = (\partial P/\partial u) \times (\partial P/\partial v)$$

  - *Remember to normalize!*
- One given by tangent of profile curve, the other by the trajectory

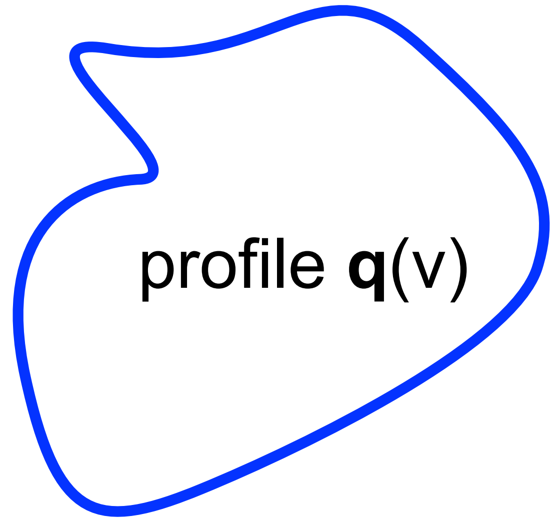


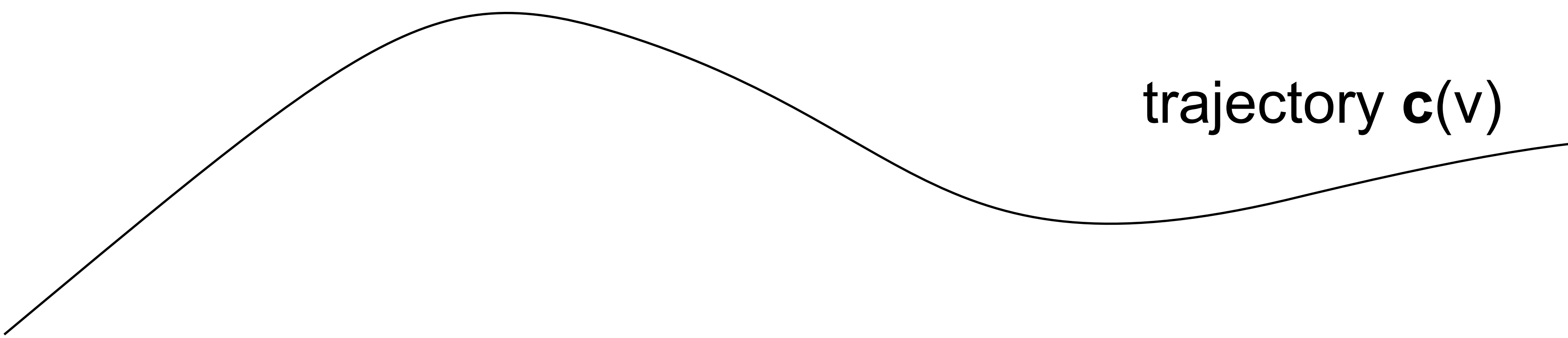
$$\boldsymbol{s}(u,v)=\boldsymbol{M}(\boldsymbol{c}(v))\boldsymbol{q}(u)$$

where ***M*** is a matrix that depends on the trajectory ***c***

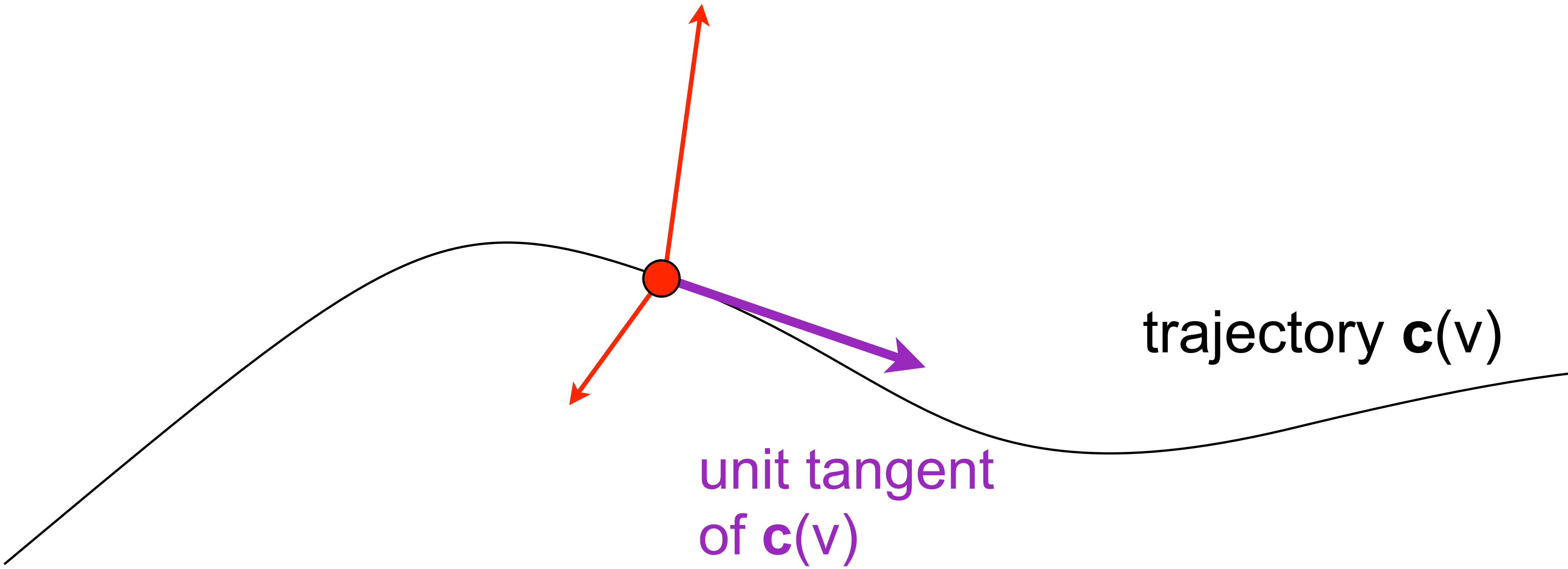
Normals for Swept Surfaces

Recommended Extra for Assignment 2
Here for your convenience, will not
cover in class.

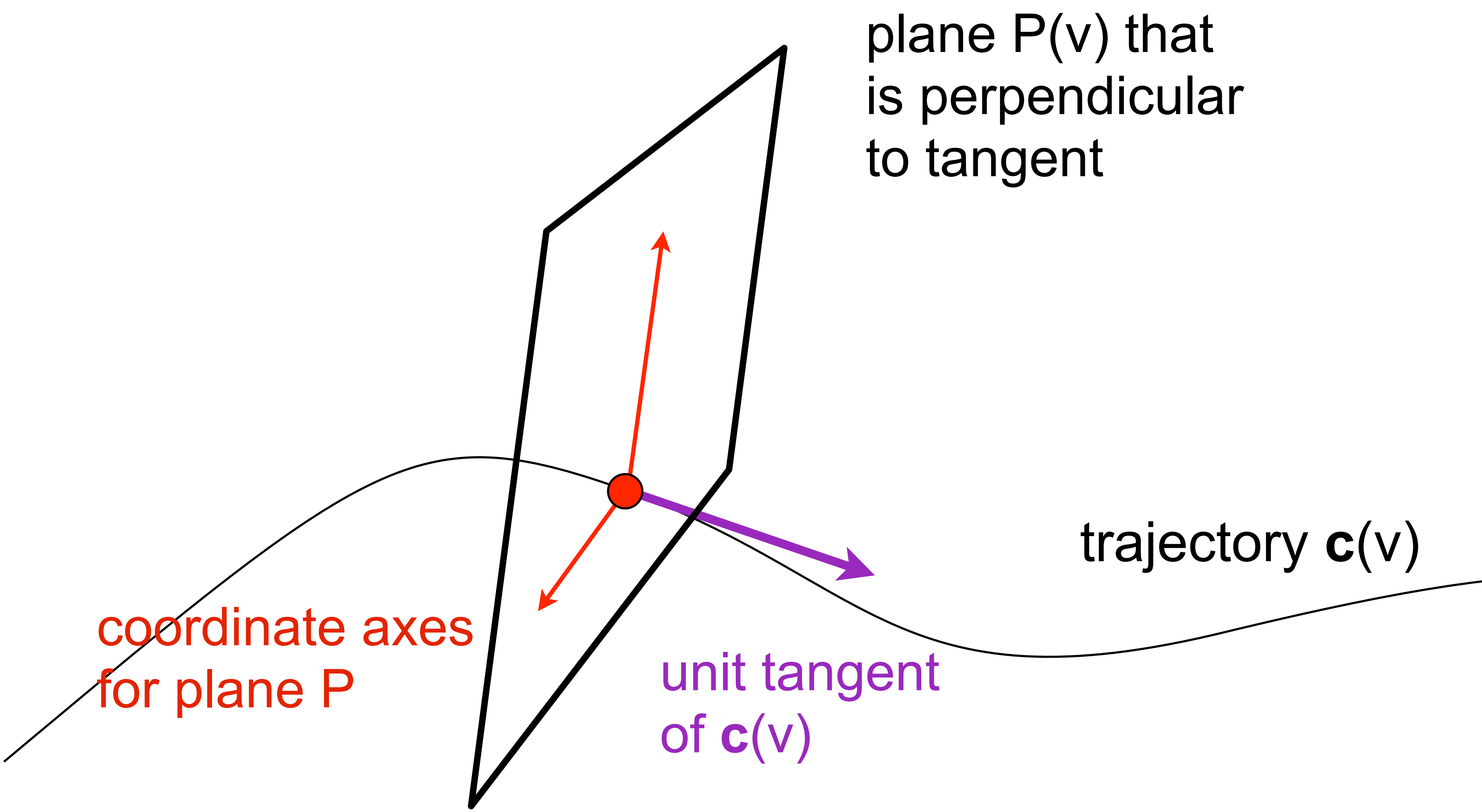




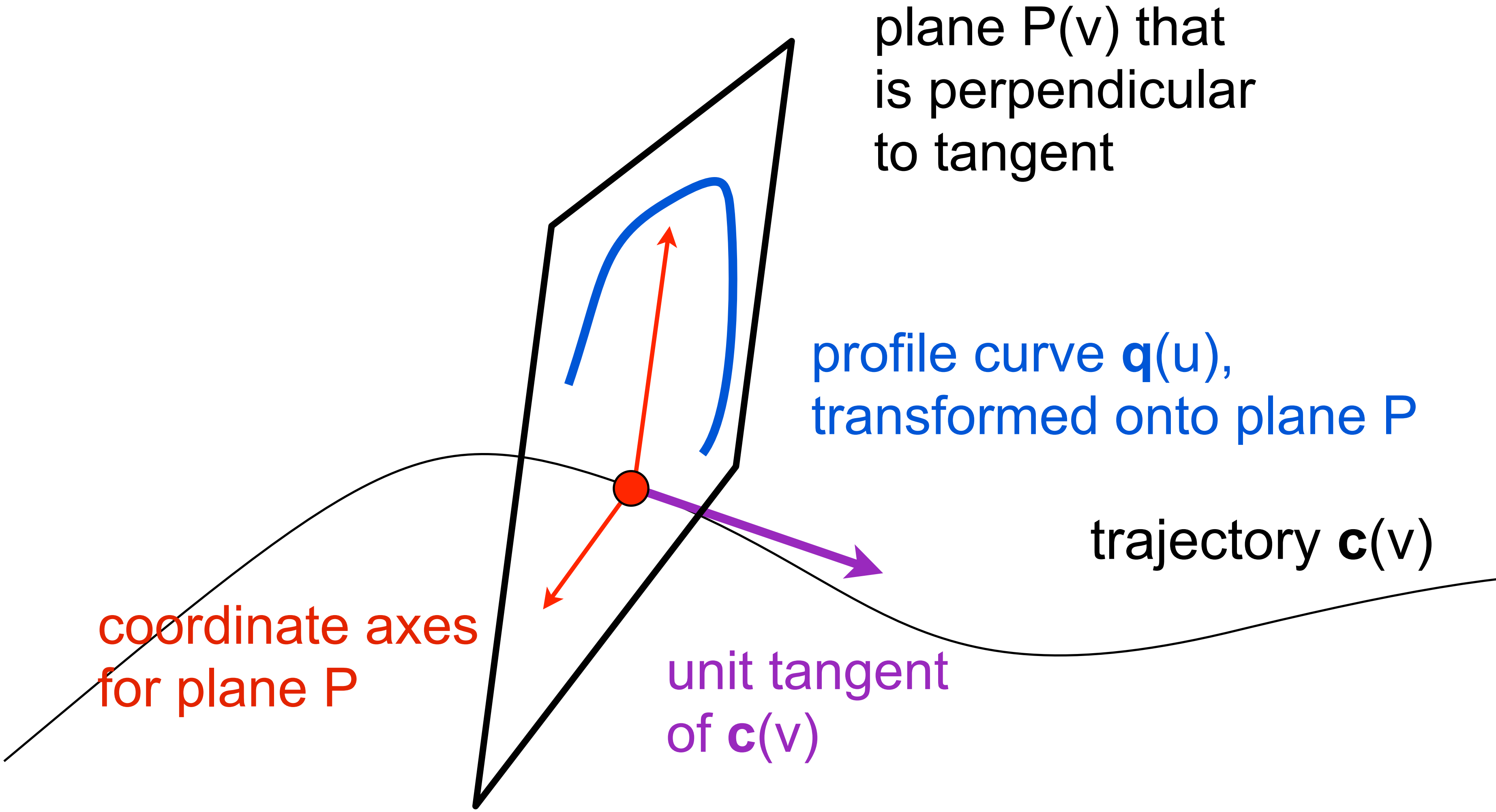
Normals for Swept Surfaces



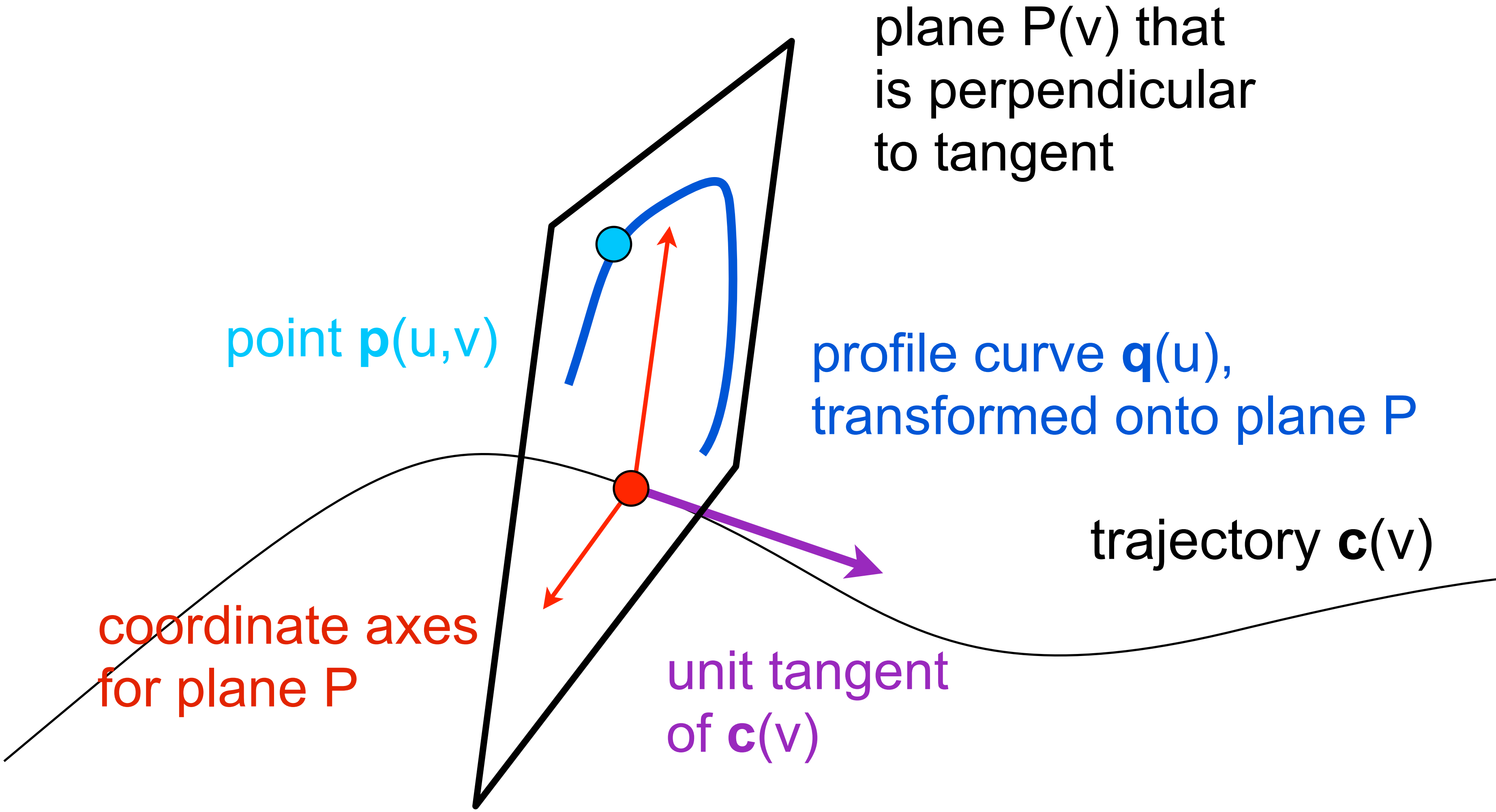
Normals for Swept Surfaces



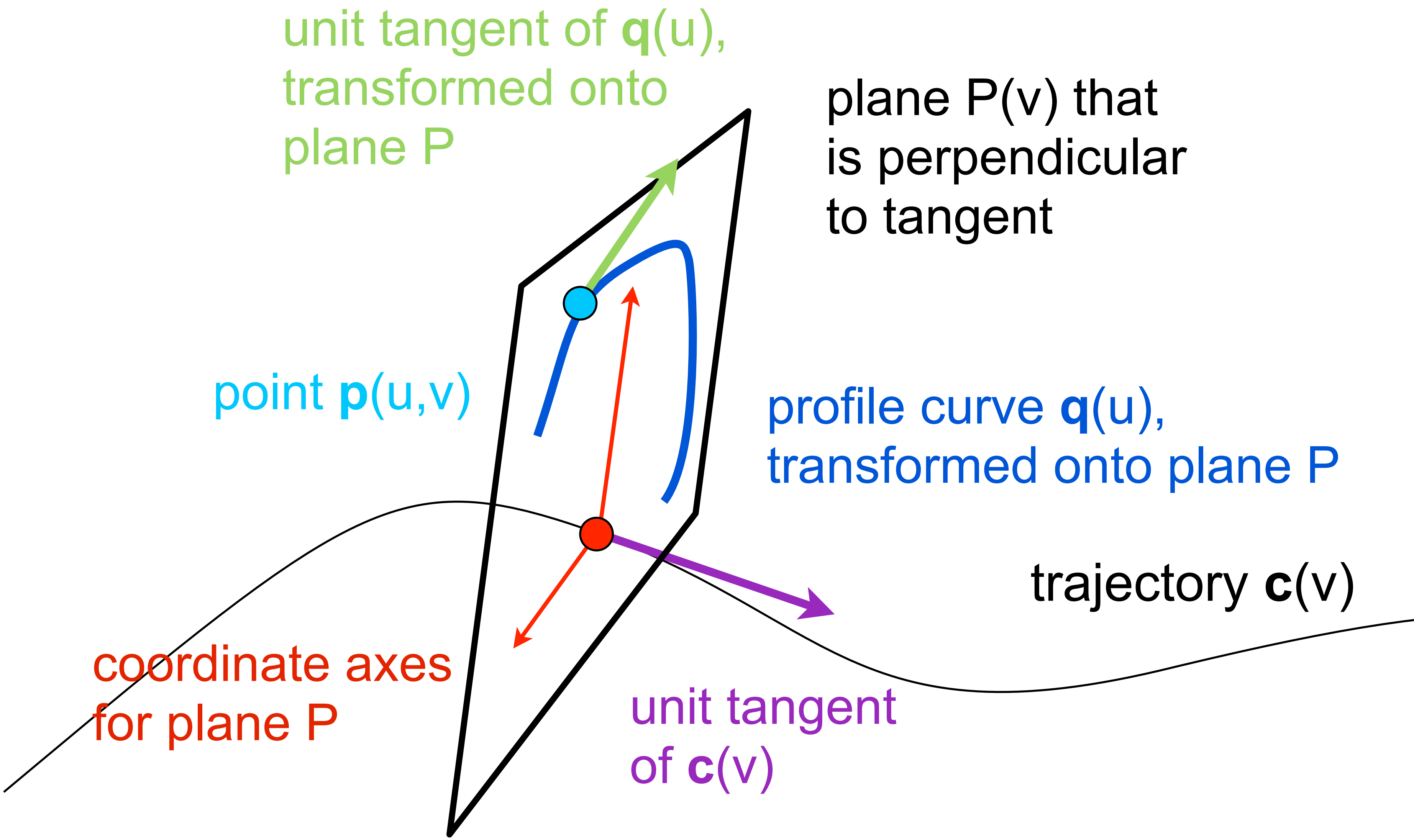
Normals for Swept Surfaces



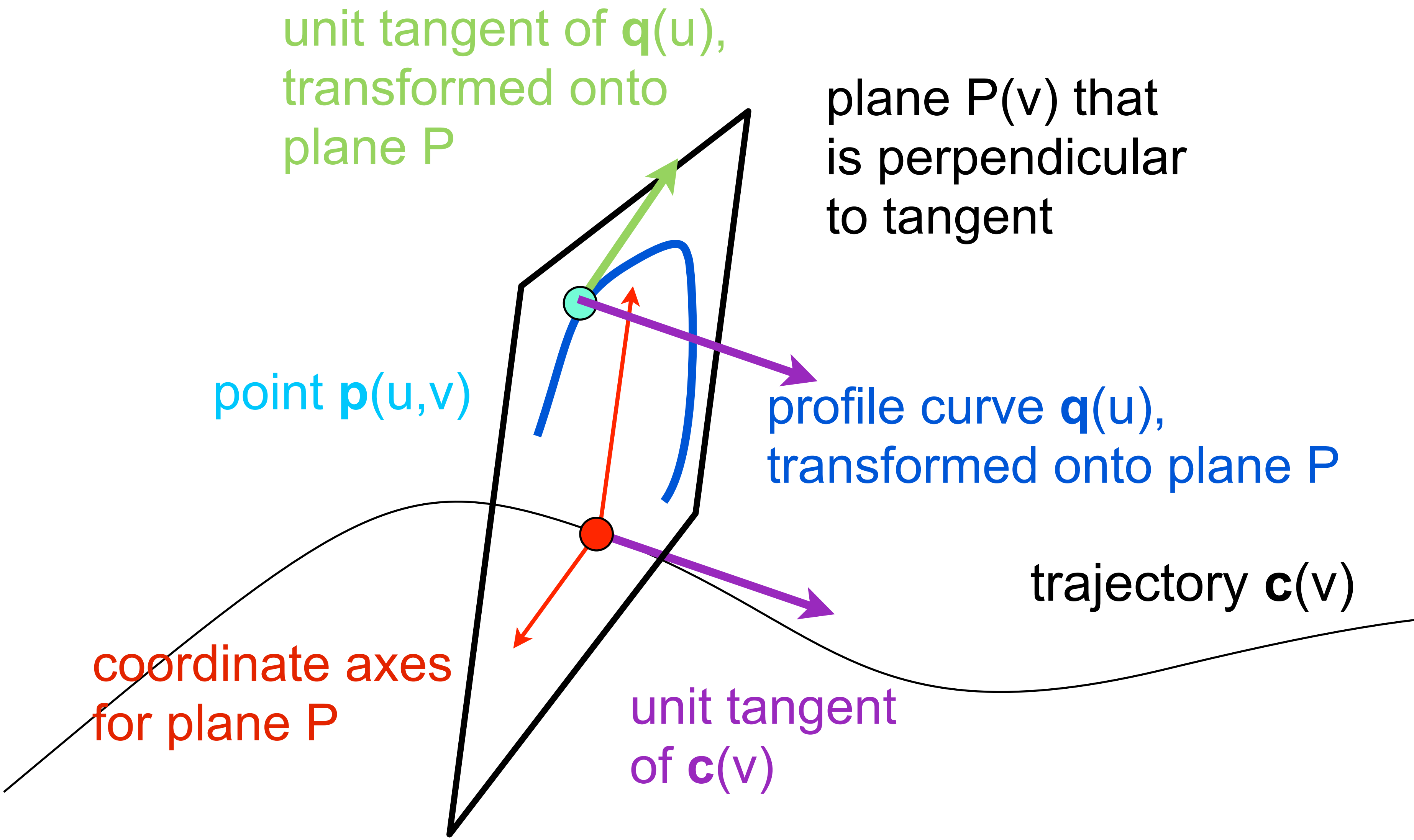
Normals for Swept Surfaces



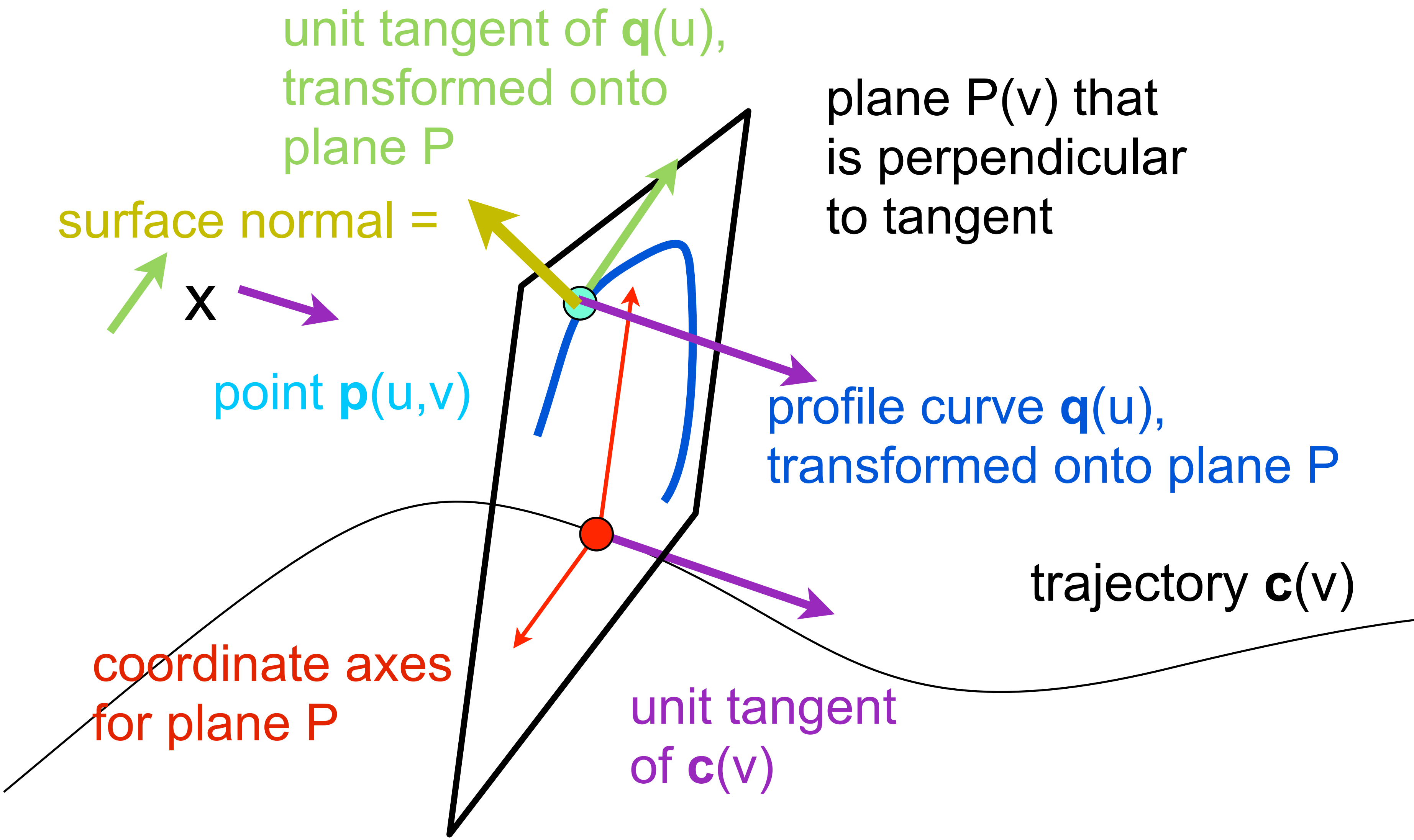
Normals for Swept Surfaces



Normals for Swept Surfaces

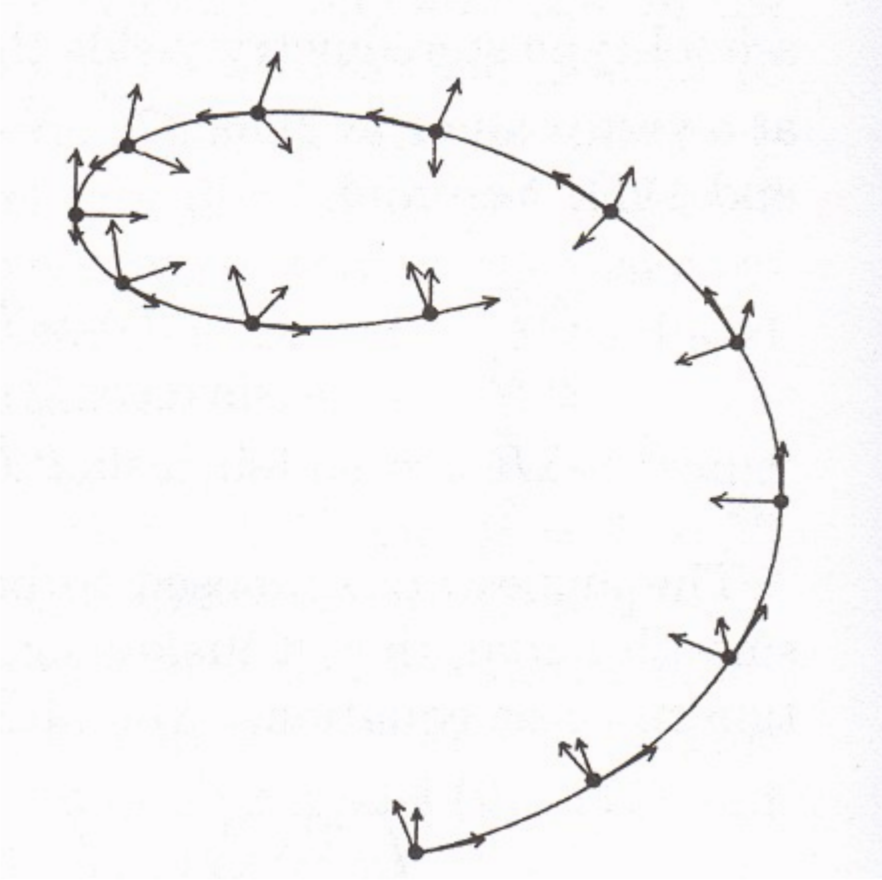


Normals for Swept Surfaces



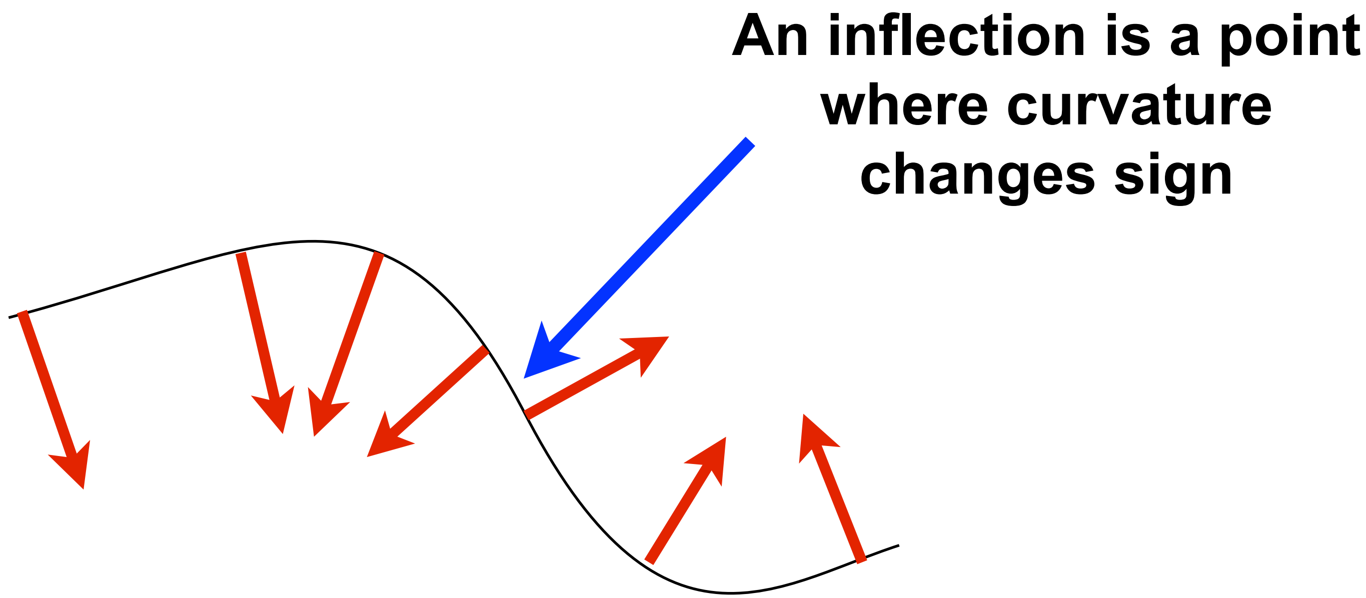
Frames on Curves: Frenet Frame

- Frame defined by 1st (tangent), 2nd (curvature) and 3rd (torsion) derivatives of a 3D curve
- Looks like a good idea for swept surfaces...

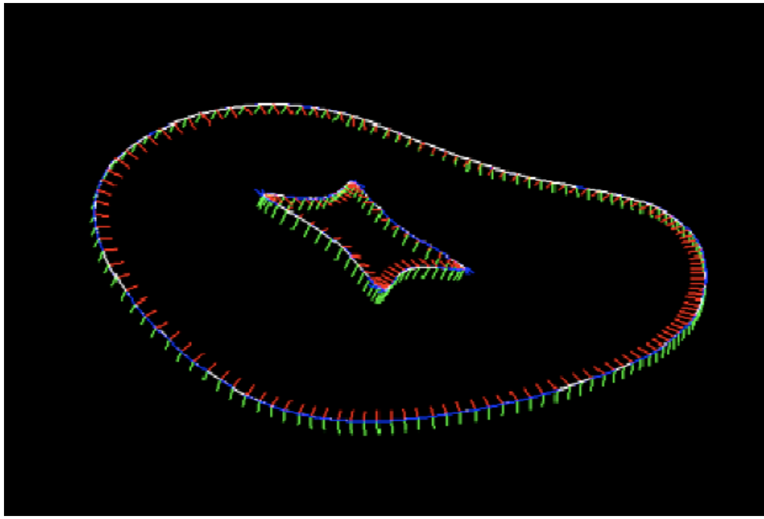


Frenet: Problem at Inflection!

- Normal flips!
- Bad to define a smooth swept surface



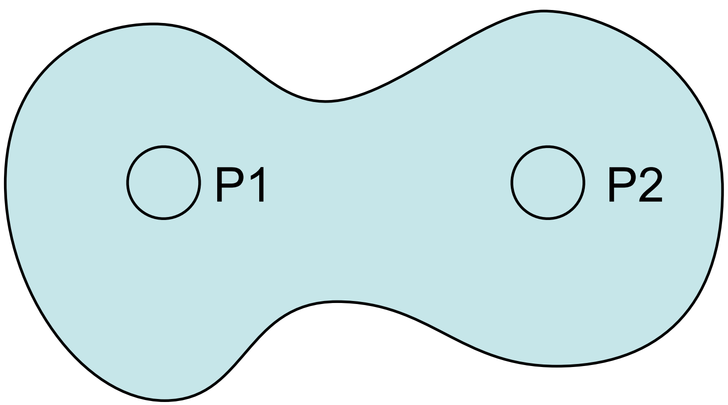
Smooth Frames on Curves

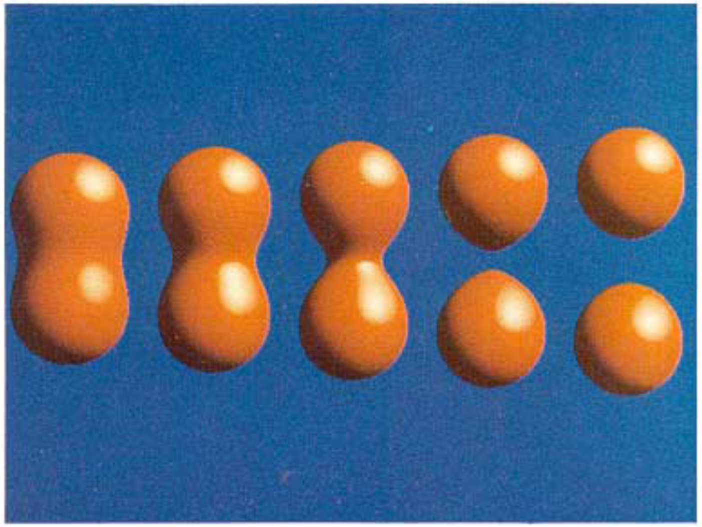


- Tangent is assumed reliable
- Build triplet of vectors
  - include tangent
  - orthonormal
  - coherent over the curve
- Idea:
  - use cross product to create orthogonal vectors
  - exploit discretization of curve
  - use previous frame to bootstrap orientation
  - **See Assignment 1 instructions!**

Implicit Surfaces

- Implicit definition: $f(x,y,z)=0$
  e.g. for a sphere: $x^2+y^2+z^2=R^2$
- Often defined as "metaballs" with seed points
- $f(x,y,z)=f1(x,y,z)+f2(x,y,z)+...$
  – where fi depends on distance to a seed point Pi

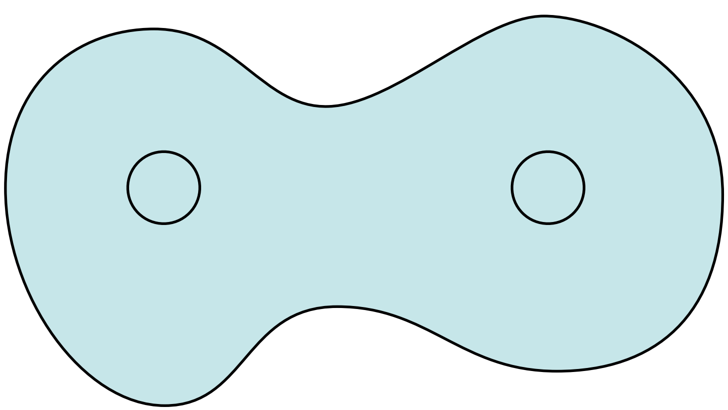


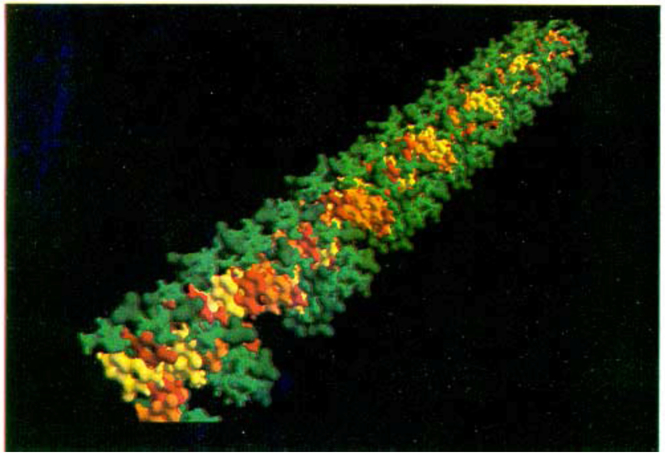


From Blinn 1982

Implicit Surfaces

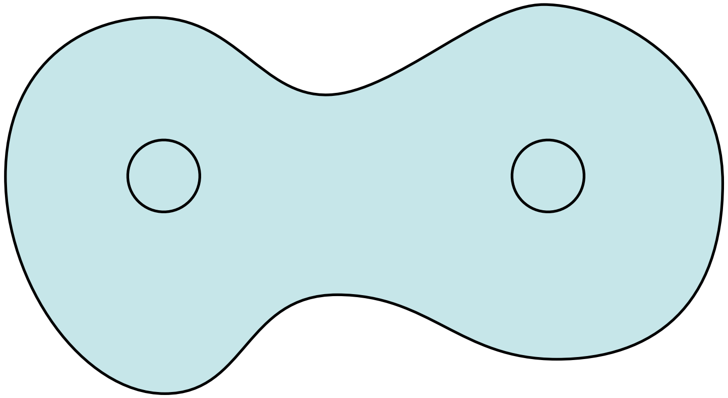
- Pros:
  - Can handle weird topology for animation
  - Easy to do sketchy modeling
  - Some data comes this way (medical & scientific data)
- Cons:
  - Does not allow us to easily generate a point on the surface

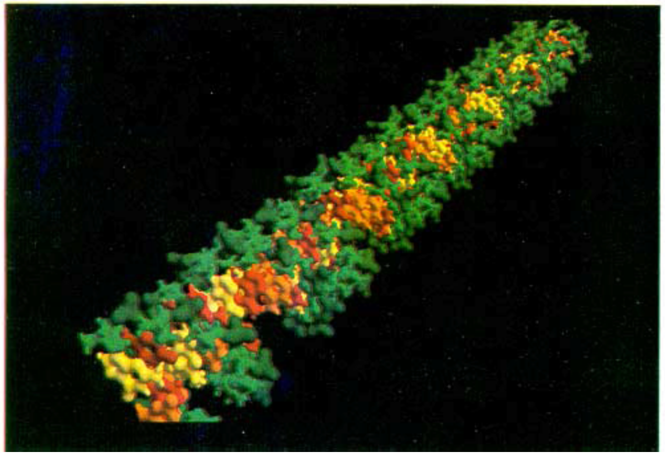




Implicit Surfaces

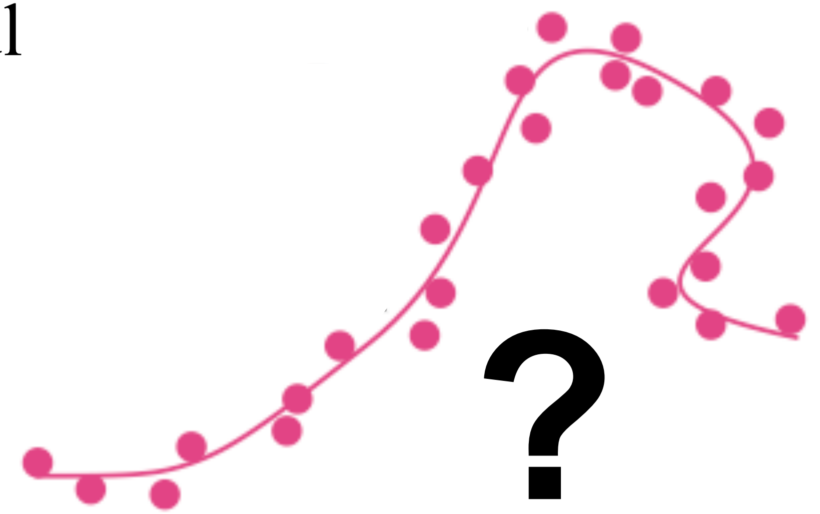
- Most common method to generate mesh from isosurface: Marching Cubes (see link for details)





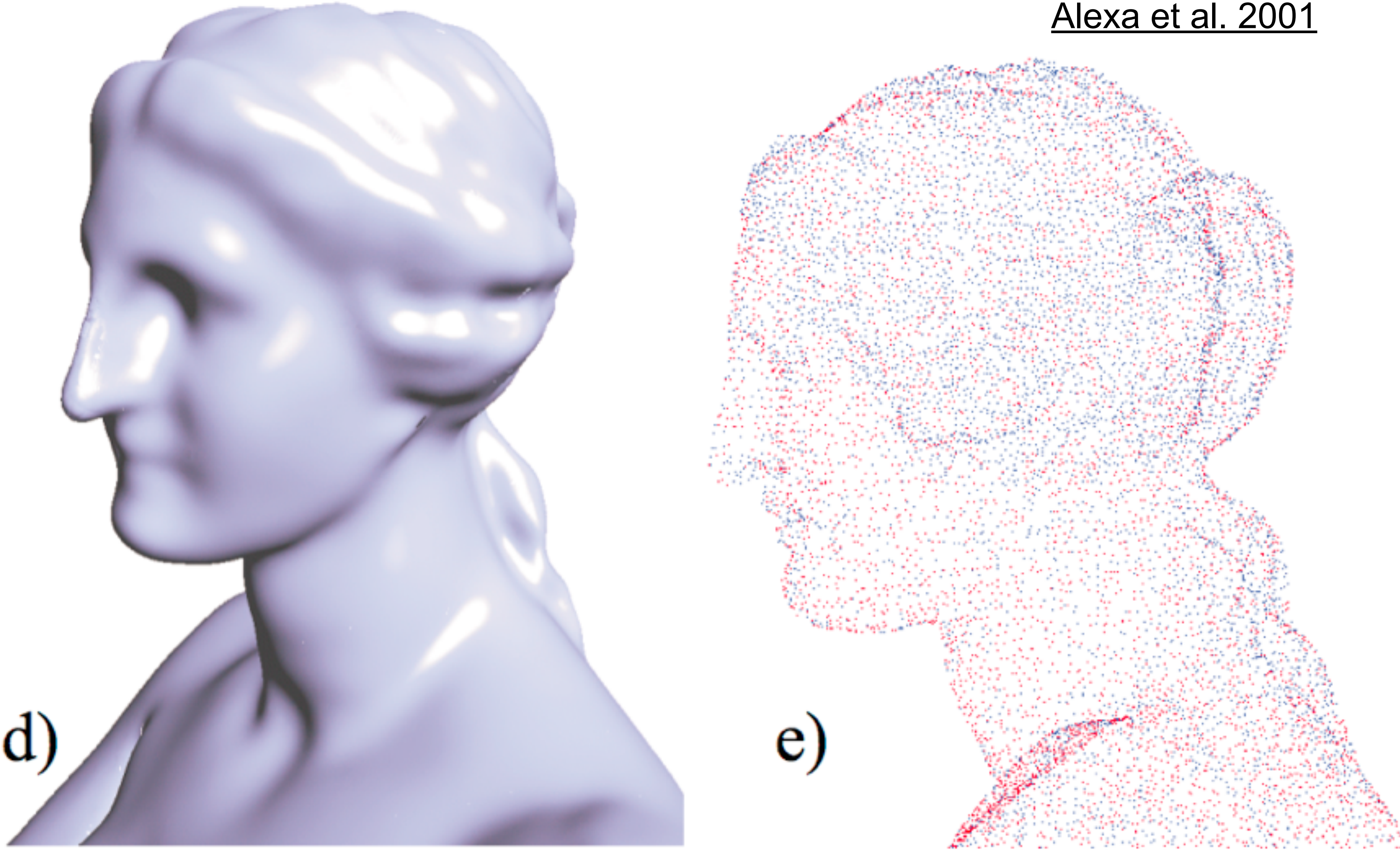
Point Set Surfaces

- Given only a noisy 3D point cloud (no connectivity), can you define a reasonable surface using only the points?
  - Laser range scans only give you points, so this is potentially useful



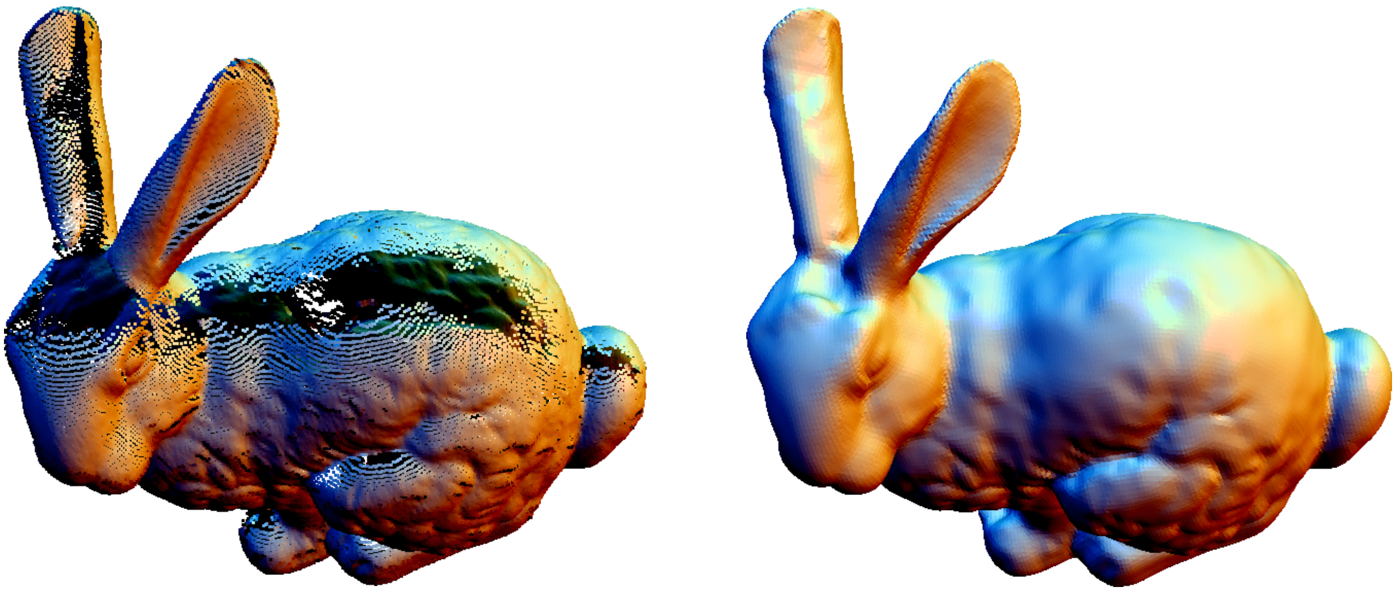
Point Set Surfaces

Alexa et al. 2001



Point Set Surfaces

- Modern take on implicit surfaces
- Cool math: Moving Least Squares (MLS), partitions of unity, etc.



Ohtake et al. 2003

- Not required in this class, but nice to know.