Real-life problems - Curves

- **Reconstruction**: given (many) points sampled from smooth curve find B-spline/NURBS/Bezier curve that interpolates/approximates input points
  - use A LOT less control than input points
  - optimize “fairness” (e.g. integral squared curvature)
  - support noise/outliers
Real-life problems - Curves

• **Editing**: given B-spline/NURBS/Bezier curve provide simple way for user to deform it
  – *allow to pick random point on curve and pull it so the rest follows in intuitive manner*

• **Modeling**: intuitive way to form desired curves
  – *more “user-friendly” than control-point editing*

Syllabus

**Curves in 2D and 3D**

**Properties of Curves and Surfaces**

**Surfaces**

• Sweeping, extrusion, surfaces of revolution
• Tensor-product surfaces
• Bézier triangles
Surfaces

Categories:
- Explicit \( z = F(x, y) \)
- Implicit \( F(x, y, z) = 0 \)
- Parametric \( F(s, t) : \mathbb{R}^2 \rightarrow \mathbb{R}^3 \)

\[
F(s, t) := \begin{pmatrix} F_x(s, t) \\ F_y(s, t) \\ F_z(s, t) \end{pmatrix}
\]

From Curves to Surfaces

Motivation
- Derive surface shapes from curves through set of simple operations
  - Extrusion
  - Revolution
  - Sweep
  - Ruling
- The curves can be modeled using any of the techniques we have discussed
**Basic surfaces**

**Concept:**
- Move a curve ("profile") along a line segment
- The union of all points visited defines the surface

**Extrusion**

\[ S(u, v) = F(u) + P_1P_2 v \]
Extrusion as Parametric Surface

**Curves:**
- Profile: arbitrary curve $P(t)$
- Line segment for sweeping:
  $$L(s) = (1 - s) \cdot p_0 + s \cdot p_1$$

**Swept Surface:**
$$F(s, t) = L(s) + P(t)$$

- Parametric function has very specific structure
- Generalized sweeping along arbitrary curve more complex

Surfaces of Revolution

**Concept:**
- Rotate profile curve around an axis
- $R(v)$ rotation matrix ($v$ in $[0, 2\pi]$)

$$S(u, v) = R(v)F(u)$$
Sweeping

Concept:
• Generalize extrusion & revolution - sweep along arbitrary curve
• To orient profile at any point
  – user specified
  – use Fresnet frame

Bilinear Patches

Bilinear interpolation of 4 3D points - 2D analog of 1D linear interpolation between 2 points in the plane
Given \( P_{00}, P_{01}, P_{10}, P_{11} \) - associated parametric bilinear surface for \( u, v \in [0,1] \) is:

\[
P(u,v) = (1-u)(1-v)P_{00} + (1-u)vP_{01} + u(1-v)P_{10} + uvP_{11}
\]

Questions:
• What does an isoparametric curve of a bilinear patch look like?
• When is a bilinear patch planar?
Bilinear Patch

Quad Mesh: Union of Bilinear Patches
Ruled Surfaces

• Given two curves $a(t)$ and $b(t)$ corresponding ruled surface is constructed by connecting curves with straight lines

$$ S(u,v) = va(u) + (1-v)b(u) $$

Questions:
• When is a ruled surface a bilinear patch?
• When is a bilinear patch a ruled surface?
Boolean Sum/Coons Patch (1967)

Given four connected curves $C_i$ $i=1, 2, 3, 4$ Boolean sum $S(u, v)$ fills the interior with surface

$S_1(u, v) = vC_1(u) + (1-v)C_3(u)$
$S_2(u, v) = uC_2(v) + (1-u)C_4(v)$
$P(u, v) = (1-u)(1-v)P_{00} + (1-u)vP_{01} + u(1-v)P_{10} + uvP_{11}$

$S(u, v) = S_1(u, v) + S_2(u, v) - P(u, v)$

$S(u, v)$ coincides with $C_i$ along its boundaries

Examples
Summary

- Simple methods for generating surfaces from curves
- Curves can be modeled any way we want
- Limited set of shapes

Clicker question

**What kind of surface best describes the pot on the right?**

A. Extrusion
B. Revolution
C. Ruled
D. Sweep
Clicker question

What kind of surface best describes the hose on the right?

A. Extrusion
B. Revolution
C. Ruled
D. Sweep

Tensor Product Surfaces

More General Parametric Surfaces
- Use basis functions like for curves
- Apply independently to parametric directions s and t
- Works for arbitrary basis

Example:
- Bézier curve:
  \[ F(t) = \sum_{i=0}^{m} B_i^m(t) \cdot b_i \]
- Tensor product Bézier patch:
  \[ F(s, t) = \sum_{i=0}^{m_i} \sum_{j=0}^{m_j} B_i^{m_i}(s) \cdot B_j^{m_j}(t) \cdot b_{i,j} \]
Tensor Product Surfaces

**Notes:**
- Surface is (rational) polynomial in $s$ and $t$, depending on basis
  - *The degree in* $s$ *is* $m_s$
  - *The degree in* $t$ *is* $m_t$
  - *The total degree is* $m_s + m_t$
- Algorithms from curves transfer directly to tensor product surfaces
- Properties of surfaces directly related to properties of corresponding curves

Tensor Product Surfaces

**Properties (Bézier, B-Spline, Rational Bézier/B-Spline):**
- *(Local)* Convex hull
- Affine invariance
- Control points of the edge curves are the boundary points of the control mesh
- Bézier patch interpolates corner vertices of its control mesh
Tensor Product Surfaces

Continuity

• Two patches

\[ F(s, t) : [s_0, s_1] \times [t_0, t_1], \]
\[ G(s, t) : [s_1, s_2] \times [t_0, t_1] \]

• are C^k continuous if for all \( t \)

\[ F^{(l)}(s, t) = G^{(l)}(s, t); l \leq k \]

• Same for \( s \)
• Special case – two patches sharing one corner

Tensor Product Surfaces

Limitations:

• Total degree is sum of degrees in \( s \) and \( t \)

• Always parameterized over rectangular parameter interval

• Refinement (degree elevation or knot insertion) always affects a whole row or column
Tensor Product Surfaces

Limitations: “suitcase corners”

Patch Networks
Clicker question

What kind of surface best describes the shape on the right?

A. Extrusion
B. Revolution
C. Sweep
D. Coons Patch

Clicker question

What kind of surface best describes the shape on the right?

A. Extrusion
B. Revolution
C. Sweep
D. Coons Patch