Deformation

- Beyond rigid body motion
  (e.g. translation, rotation)

- Extremely valuable for
  both modeling and rendering

- Applications
  - animation, design, visualization
  - engineering, medicine
  - education, simulation

- Techniques

- Geometric manipulation via control points

- Dynamic sculpting via physical forces

- Local/global deformation

- Hierarchical deformation

- **Space deformation**
- **Free-form deformation**
- **Deformation control**
- **Examples: twisting, bending, tapering**

Space Deformation

- Coordinate axis deformation (e.g. x-axis)
- Linear/nonlinear deformation
- Arbitrary dimension

$$\mathbf{c}(u) = \begin{bmatrix} x(u) \\ y(u) \\ z(u) \end{bmatrix} =$$

$$\begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} x(u) + \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} y(u) + \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} z(u)$$

$$= \mathbf{x} + \mathbf{y} + \mathbf{z}$$

where

$$\mathbf{x} = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} x(u)$$

$\mathbf{y}$ are $\mathbf{z}$ are obtained in a similar way

- Deform x-axis by a new curve $\mathbf{r}(x)$

- **The new deformed curve**

$$\mathbf{c}'(u) = \mathbf{r}(x(u)) + y(u)\mathbf{m} + z\mathbf{b}$$

where $(\mathrm{t}, \mathrm{m}, \mathrm{b})$ **is a Frenet frame at** $x(u)$ **on** $\mathrm{r}(x(u))$

- **Many choices of coordinate system are available**

$$\mathbf{c}(u) = \mathbf{r}(u(x)) + \mathbf{q}(y(u)) + \mathbf{p}(z(u))$$

- **Non-orthogonal system (change angles)**

$$\mathbf{c}(u) = x(u)\mathbf{i}' + y(u)\mathbf{j}' + z(u)\mathbf{k}'$$

- **Other algorithms**

Line (Wire) Deformation

- Generalization of coordinate-axis deformation
- Reference line (wire) is attached to the model
- Deformation of reference lines (wires)
- Subsequent deformation is propagated to the surrounding (localized) environment
- Scaling, bending, twisting, and stretching
- Straight line (or curved wire)
- Spline-based curves
- The key: how neighborhood points are defined and deformed based on local coordinate systems
- Example: Frenet frame
- Univariate deformation

- **Generalize to bivariate/trivariate deformation**

Bivariate/Trivariate Deformation

- **Undeformed space (2D or 3D) defined by bivariate surface** $\mathrm{s}(u, v)$
- **Deform** $\mathrm{s}(u, v)$
- **Deform the object (within or nearby) the space**
- **Curve:** $\mathbf{c}(t) = [u(t), v(t)]^\top$
- **New curve:** $\mathrm{s}(u(t), v(t))$
- **Local coordinate system of deformation space**
- **Space** $\mathrm{s}(u, v)$ **can be**
  - **Hermite, Bezier, B-spline, NURBS, etc.**
- **Generalization to 3D Solid**
- **Solid space:** $\mathrm{s}(u, v, w)$
- **Curve:** $\mathbf{c}(t) = [u(t), v(t), w(t)]^\top$

- **New curve:** $\mathrm{s}(u(t), v(t), w(t))$
- **Applicable to surface/volume deformation**
- **General procedure**
  - **arbitrary objects in space**
  - **deform the space (coordinate system)**
  - **object deformation**
- **Free-form deformation (predictability)**
- **Algebraic deformation (apply algebraic functions)**

Free-Form Deformation

- Space defined by free-form splines
- Objects are embedded in (attached to) space
- Spline deformation (space deformation)
- Objects are deformed
- Bivariate deformation (2D, curve)
- Trivariate deformation (3D, curve, surface, volume)
- Bezier, B-spline, NURBS space
- Subdivision-based space
- Sweeping space
- Other more complicated space
- Not just restricted to geometric domain

- **Applicable to texture, color, intensity, material, etc.**

Modeling and Deformation

- From simpler primitives to complex forms using operators
  (e.g. CSG)
  - rigid body transformation
  - Boolean operations
- Deformation is a modeling tool!
- Complex objects from simpler ones
- Generalization of the conventional operations
- Examples: bend, twist, taper, compress, expand, etc.
- Intuitive and easy to control
- Simulate natural manufacturing process
- New shape modeling operations
- Hierarchical structure for complicated objects

- **Computation on deformed objects should be easy (derived from non-deformed objects)**
  - **fast rendering (position, tangent, normal)**
  - **interaction (spatial occupancy, collision detection)**
- **Parameterization can be viewed as deformation**

$$\mathbf{c}(u)$$

$$\mathbf{s}(u, v)$$

$$\mathbf{s}(u, v, w)$$

Deformation Examples

- **Scaling**

$$x' = a_1 x$$

$$y' = a_2 y$$

$$z' = a_3 z$$

- **z-axis tapering**

$$r = f(z)$$

$$x' = rx$$

$$y' = ry$$

$$z' = z$$

- **z-Axis twisting**

$$\theta = f(z)$$

$$x' = x \cos\theta - y \sin\theta$$

$$y' = x \sin\theta + y \cos\theta$$

$$z' = z$$

Bending (y-axis)

- **The bending is from y to z**
- **The bending happens between** $y = a$ **and** $y = b$
- **The bending rate (curvature, radians per unit length)** κ
- **The bending angle (in radians) is** θ **(w.r.t. z-axis)**

$$\theta = \begin{cases} 0 & y <= a \\ \kappa(y-a) & a < y < b \\ \kappa(b-a) & b <= y \end{cases}$$

- **Radius of curvature**

$$\rho = \frac{1}{\kappa}$$

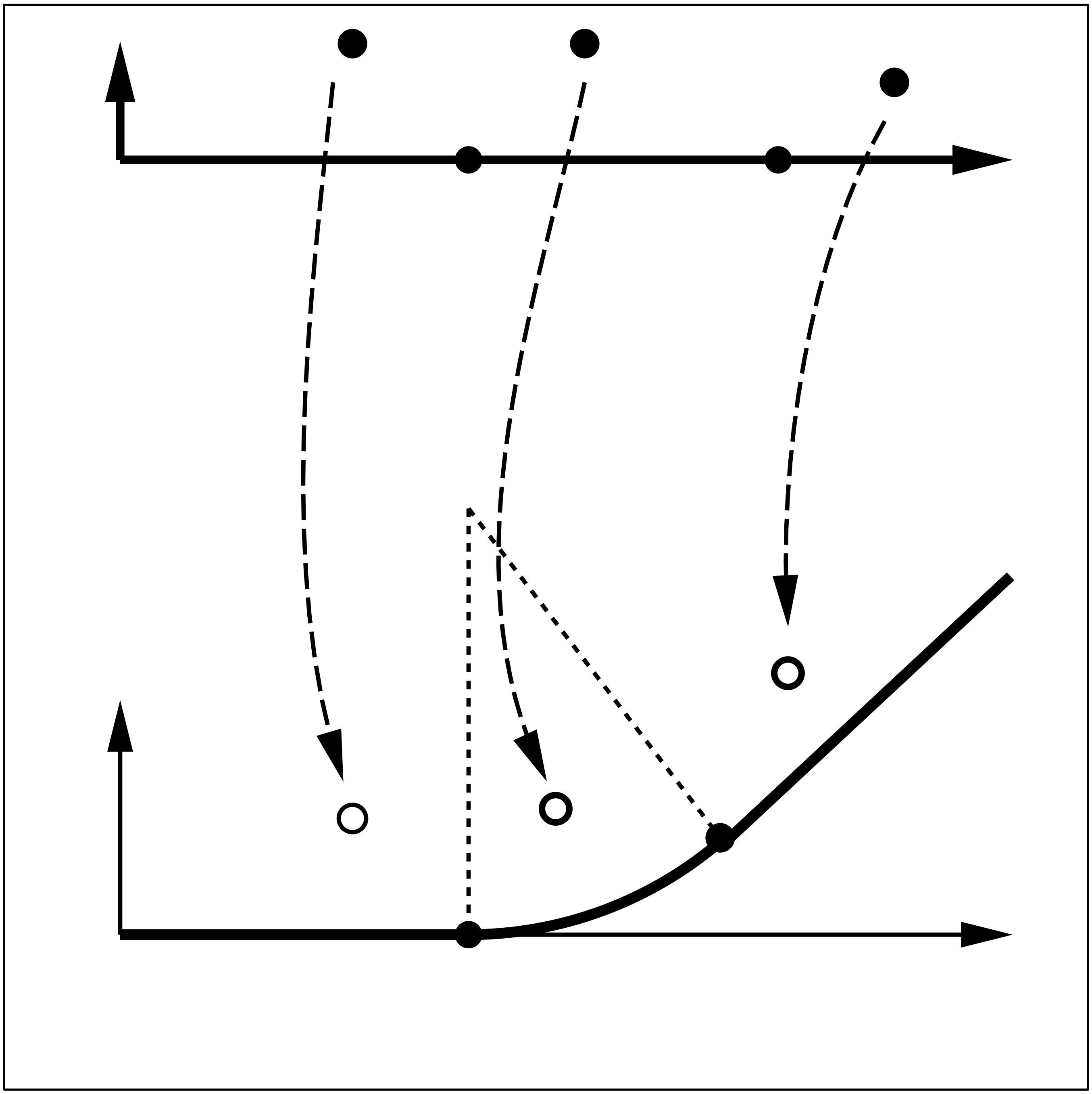
- **After the bending deformation**

$$x' = x$$

$$y' = \begin{cases} y & y < a \\ a - (\sin\theta)(z-\rho) & a <= y <= b \\ a - (\sin\theta)(z-\rho) + (\cos\theta)(y-b) & b < y \end{cases}$$

$$z' = \begin{cases} z & y < a \\ \rho + (\cos\theta)(z - \rho) & a <= y <= b \\ \rho + (\cos\theta)(z - \rho) + (\sin\theta)(y - b) & b < y \end{cases}$$

Linear Bending



Free-Form Deformation

- Embed arbitrary object(s) within a space
- The space is defined as a free-form solid
- Deform the space (free-form deformation (FFD))
- Objects are deformed (indirectly)
- Space is defined by trivariate Bezier solid

$$\mathbf{s}_{ffd}(u,v,w) = \sum_{i=0}^{l} \sum_{j=0}^{m} \sum_{k=0}^{n} \mathbf{p}_{ijk} B_i(u) B_j(v) B_k(w)$$

- Parallelpiped region and its parameterization

$$\mathbf{x} = \mathbf{x}_0 + u\mathbf{a} + v\mathbf{b} + w\mathbf{c}$$

where (u,v,w) are parametric coordinates for $\mathbf{x}$

$$u = \frac{\mathbf{b} \times \mathbf{c} \cdot (\mathbf{x} - \mathbf{x}_0)}{\mathbf{b} \times \mathbf{c} \cdot (\mathbf{a})}$$

$$v = \frac{\mathbf{a} \times \mathbf{c} \cdot (\mathbf{x} - \mathbf{x}_0)}{\mathbf{a} \times \mathbf{c} \cdot (\mathbf{b})}$$

$$w = \frac{\mathbf{a} \times \mathbf{b} \cdot (\mathbf{x} - \mathbf{x}_0)}{\mathbf{a} \times \mathbf{b} \cdot (\mathbf{c})}$$

where $u, v, w \in [0, 1]$

- **Use trivariate Bezier solid to represent parallelpiped region**

- **Control points are regular lattice points**

$$\mathrm{p}_{ijk} = \mathrm{x}_0 + \frac{i}{l}\mathrm{a} + \frac{j}{m}\mathrm{b} + \frac{k}{n}\mathrm{c}$$

- **Deform the space (move control points** p_{ijk} **to** p'_{ijk}**)**

- **Point** x **will be moved to** x'

$$\mathrm{x} = \mathrm{s}_{ffd}(u, v, w)$$

$$\mathrm{x}' = \mathrm{s}'_{ffd}(u, v, w)$$

- **Useful for curve, surface, solid modeling**

- **Applicable to polygonal datasets**

- **Example: surface manipulation**

$$x(\alpha, \beta), y(\alpha, \beta), z(\alpha, \beta)$$

Assume (u, v, w) is the parametric value for the surface within free-form solid

$$\mathrm{s}'_{ffd}(u, v, w)$$

Generalizations

- Properties
- Arbitrary objects (curves, surfaces, solids)
- Intuitive and natural, but indirect sculpting
- Continuity control

$$\mathrm{s}_1(u_1, v_1, w_1)$$

$$\mathrm{s}_2(u_2, v_2, w_1)$$

  One part is embedded in s_1, an adjacent part is embedded in s_2

- Multiple spaces
- Local/global deformation
- Volume preserving deformation (how to evaluate the volume)

- **B-spline solid**
- **NURBS solid**
- **From regular lattice (parallelepiped) to cylindrical latt**
- **Prismatic lattice**
- **Tetrahedron-based lattice**
- **Lattices of arbitrary topology**
- **FFD as a animation tool**
  - **metamorphosis**
- **Allow FFD space to move w.r.t. object space**
- **Direct manipulation of object space**