

Curves

- Hermite Curve
- Bézier Curve
- B-Spline Curve
- Nonuniform Rational B-Spline (NURBS) Curve

$$\mathbf{U} = \begin{bmatrix} 1 & u & u^2 & u^3 \end{bmatrix}, \mathbf{A} = \begin{bmatrix} \mathbf{a}_0 \\ \mathbf{a}_1 \\ \mathbf{a}_2 \\ \mathbf{a}_3 \end{bmatrix}, \mathbf{S} = \begin{bmatrix} \mathbf{P}_0 \\ \mathbf{P}_1 \\ \mathbf{t}_0 \\ \mathbf{t}_1 \end{bmatrix}, \mathbf{R} = \begin{bmatrix} \mathbf{V}_0 \\ \mathbf{V}_1 \\ \mathbf{V}_2 \\ \mathbf{V}_3 \end{bmatrix}$$
$$\mathbf{r}(t) = \begin{cases} \mathbf{UA} \leftarrow \text{parametric curve} \\ \mathbf{UCS} \leftarrow \text{Hermite curve} \\ \mathbf{UMR} \leftarrow \text{Bezier curve} \\ \mathbf{UNR} \leftarrow \text{Uniform B-spline curve} \end{cases}$$

Curve Model: Parametric Polynomial

- 초월함수(sin, cos, log, ...)를 쓰지 않고 다항식을 씀

$$x(t) = a_0 + a_1t + a_2t^2 + a_3t^3 + \cdots + a_nt^n$$

- 자유곡선의 표현에 적합, CAD/CAM 시스템에서 가장 널리 이용
- Parametric polynomial curve의 종류
 - Power basis Polynomial Curve
 - Ferguson (Hermite) Curve
 - Bezier Curve
 - B-Spline Curve
 - 이들은 모두 상호 변환 가능함

Parametric Cubic Curve (3차)

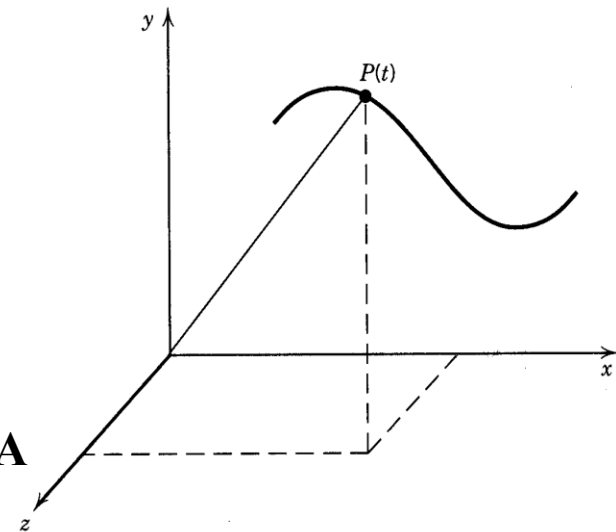
- Scalar form
$$\begin{cases} x(u) = a_{0x} + a_{1x}u + a_{2x}u^2 + a_{3x}u^3 \\ y(u) = a_{0y} + a_{1y}u + a_{2y}u^2 + a_{3y}u^3 \\ z(u) = a_{0z} + a_{1z}u + a_{2z}u^2 + a_{3z}u^3 \end{cases} \quad 0 \leq u \leq 1$$

- Vector form
$$\mathbf{r}(u) = \mathbf{a}_0 + \mathbf{a}_1u + \mathbf{a}_2u^2 + \mathbf{a}_3u^3$$

$$= \sum_{i=0}^3 u^i \mathbf{a}_i$$

- Matrix form

$$\mathbf{r}(u) = \begin{bmatrix} 1 & u & u^2 & u^3 \end{bmatrix} \begin{bmatrix} \mathbf{a}_0 \\ \mathbf{a}_1 \\ \mathbf{a}_2 \\ \mathbf{a}_3 \end{bmatrix} = \mathbf{U}\mathbf{A}$$



- 장점: 계산 속도가 빠르다.
- 단점: 계수 $[\mathbf{a}_0, \mathbf{a}_1, \mathbf{a}_2, \mathbf{a}_3]$ 의 의미를 파악하기 어렵다.

Hermite (Ferguson) Curve (1)

- Polynomial 곡선식 $\mathbf{r}(u) = \mathbf{a}_0 + \mathbf{a}_1u + \mathbf{a}_2u^2 + \mathbf{a}_3u^3 \quad (0 \leq u \leq 1)$

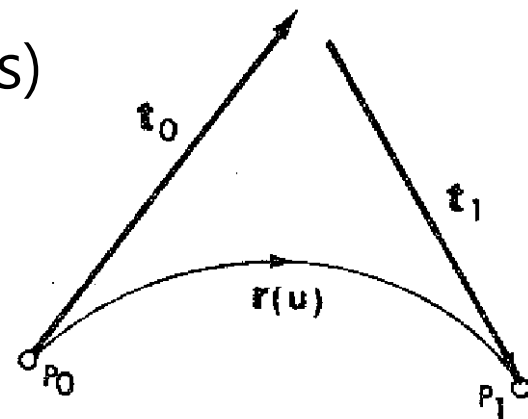
$$\mathbf{r}'(u) = \mathbf{a}_1 + 2\mathbf{a}_2u + 3\mathbf{a}_3u^2$$

- 곡선 양단에서의 끝점(P_0, P_1)과 접선벡터(t_0, t_1)를 대입

$$\begin{cases} \mathbf{r}(0) = \\ \mathbf{r}(1) = \\ \mathbf{r}'(0) = \\ \mathbf{r}'(1) = \end{cases}$$

- a_i 에 대하여 풀면 (algebraic coefficients)

$$\begin{cases} \mathbf{a}_0 = \\ \mathbf{a}_1 = \\ \mathbf{a}_2 = \\ \mathbf{a}_3 = \end{cases}$$



Hermite (Ferguson) Curve (2)

- 위의 식들을 대입하여 $\mathbf{r}(u)$ 로 표시

$$\mathbf{r}(u) = \mathbf{a}_0 + \mathbf{a}_1 u + \mathbf{a}_2 u^2 + \mathbf{a}_3 u^3 \quad (0 \leq u \leq 1)$$

=

=

geometric coefficients

$$\begin{cases} \mathbf{P}_0 = \mathbf{P}(0) \\ \mathbf{P}_1 = \mathbf{P}(1) \\ \mathbf{t}_0 = \mathbf{P}'(0) \\ \mathbf{t}_1 = \mathbf{P}'(1) \end{cases}$$

blending functions

$$\begin{cases} f_1(u) = 1 - 3u^2 + 2u^3 \\ f_2(u) = 3u^2 - 2u^3 \\ f_3(u) = u - 2u^2 + u^3 \\ f_4(u) = -u^2 + u^3 \end{cases}$$

- Matrix form

$$\mathbf{r}(u) = \begin{bmatrix} 1 & u & u^2 & u^3 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ -3 & 3 & -2 & -1 \\ 2 & -2 & 1 & 1 \end{bmatrix} \begin{bmatrix} \mathbf{P}_0 \\ \mathbf{P}_1 \\ \mathbf{t}_0 \\ \mathbf{t}_1 \end{bmatrix} = \mathbf{UCS}$$

Matrix Form

$$\mathbf{f}(t) = \mathbf{a}t^3 + \mathbf{b}t^2 + \mathbf{c}t + \mathbf{d}$$

$$\begin{bmatrix} t^3 & t^2 & t & 1 \end{bmatrix} \begin{bmatrix} \times & \times & \times & \times \\ \times & \times & \times & \times \\ \times & \times & \times & \times \\ \times & \times & \times & \times \end{bmatrix} \begin{bmatrix} \mathbf{p}_0 \\ \mathbf{p}_1 \\ \mathbf{p}_2 \\ \mathbf{p}_3 \end{bmatrix}$$

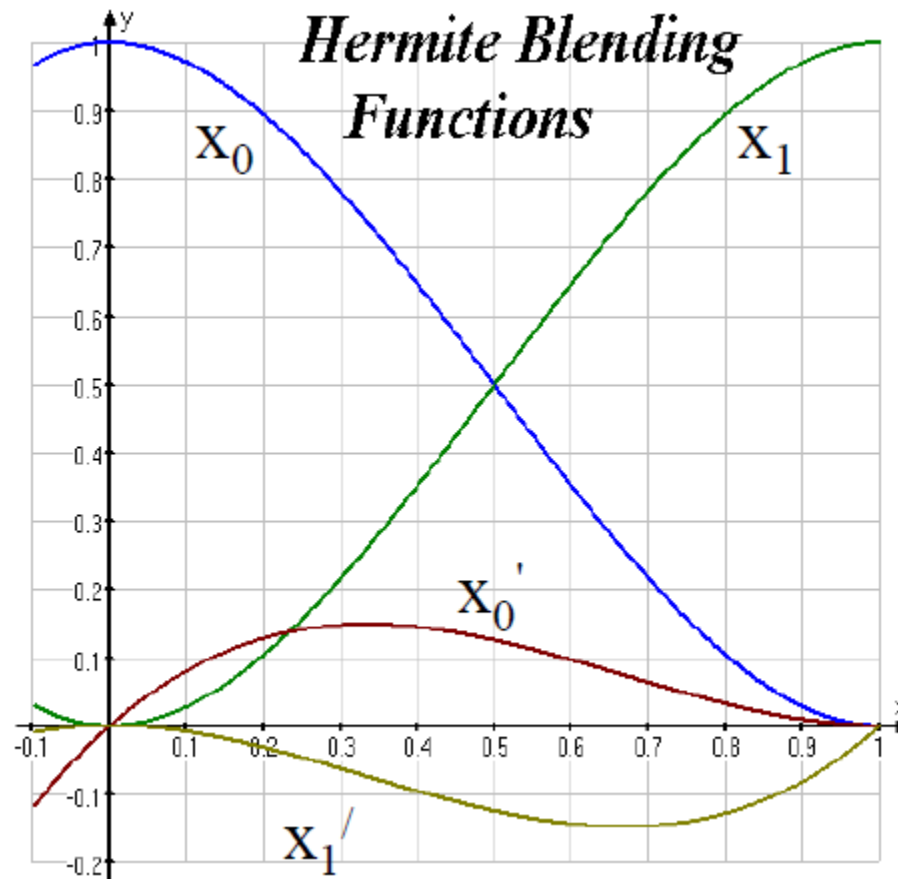
$$\begin{bmatrix} t^3 & t^2 & t & 1 \end{bmatrix} \begin{bmatrix} \times & \times & \times & \times \\ \times & \times & \times & \times \\ \times & \times & \times & \times \\ \times & \times & \times & \times \end{bmatrix} \begin{bmatrix} \mathbf{p}_0 \\ \mathbf{p}_1 \\ \mathbf{p}_2 \\ \mathbf{p}_3 \end{bmatrix}$$

$$\mathbf{f}(t) = b_0(t)\mathbf{p}_0 + b_1(t)\mathbf{p}_1 + b_2(t)\mathbf{p}_2 + b_3(t)\mathbf{p}_3$$

- coefficients = rows
- basis functions = columns

Hermite Basis (Blending) Function

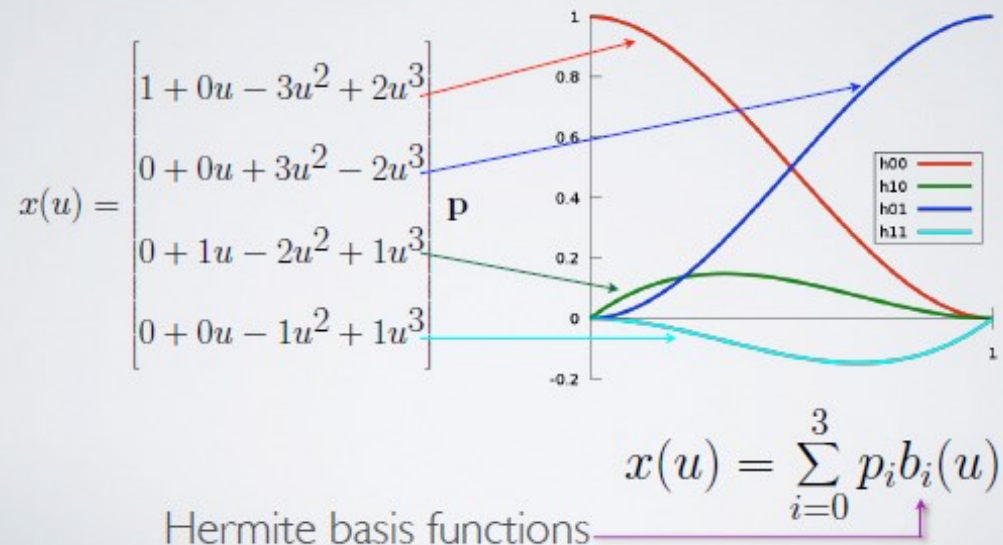
$$\underline{X(t) = (2t^3 - 3t^2 + 1)x_0 + (t^3 - 2t^2 + t)x_0' + (-2t^3 + 3t^2)x_1 + (t^3 - t^2)x_1'}$$



Cubic Hermite Basis

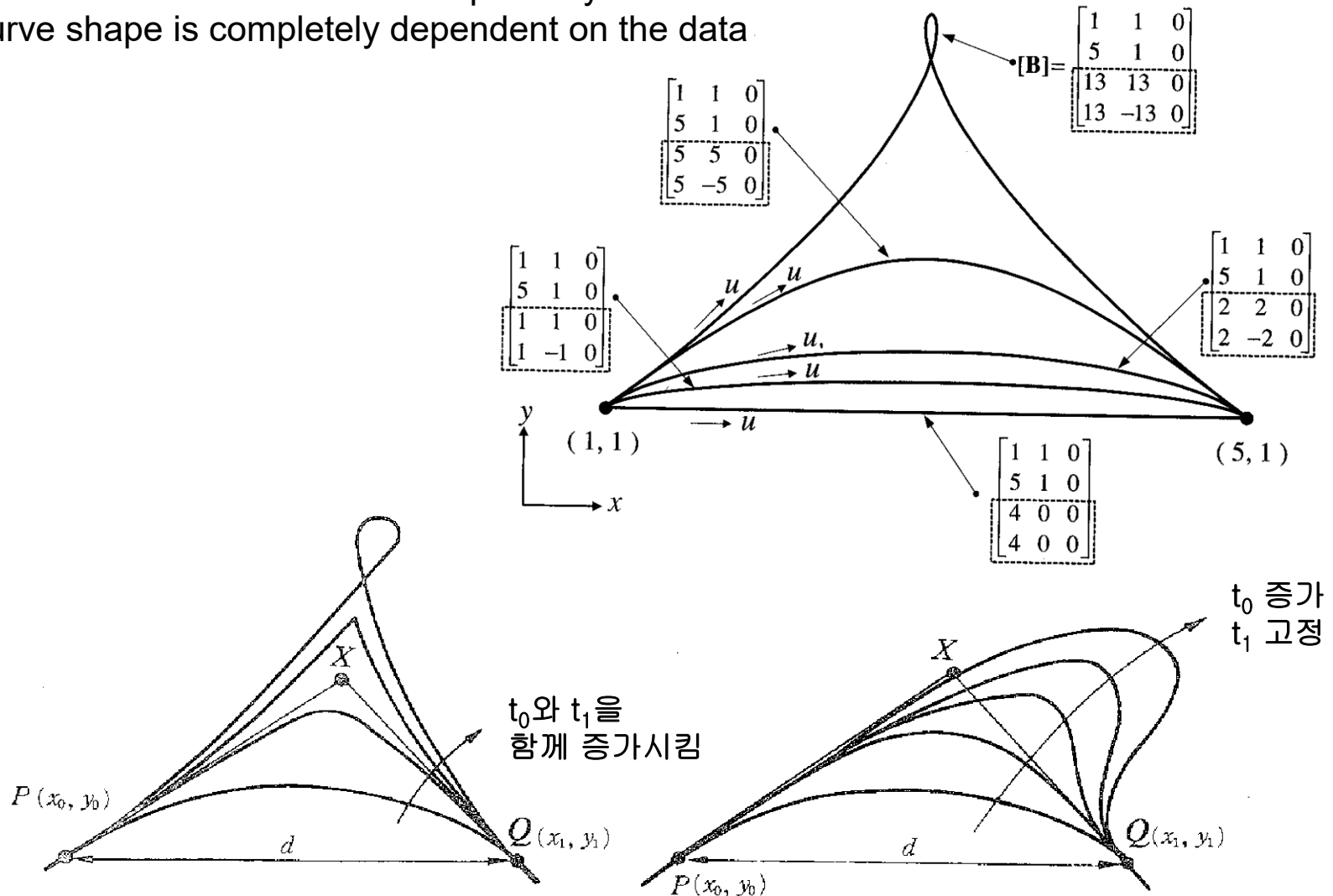
- Specify curve by
 - Endpoint values
 - Endpoint tangents (derivatives)
- Parametric interval is arbitrary
 - Don't need to recompute basis functions

Given desired values (constraints) how do we determine the coefficients for cubic power basis?



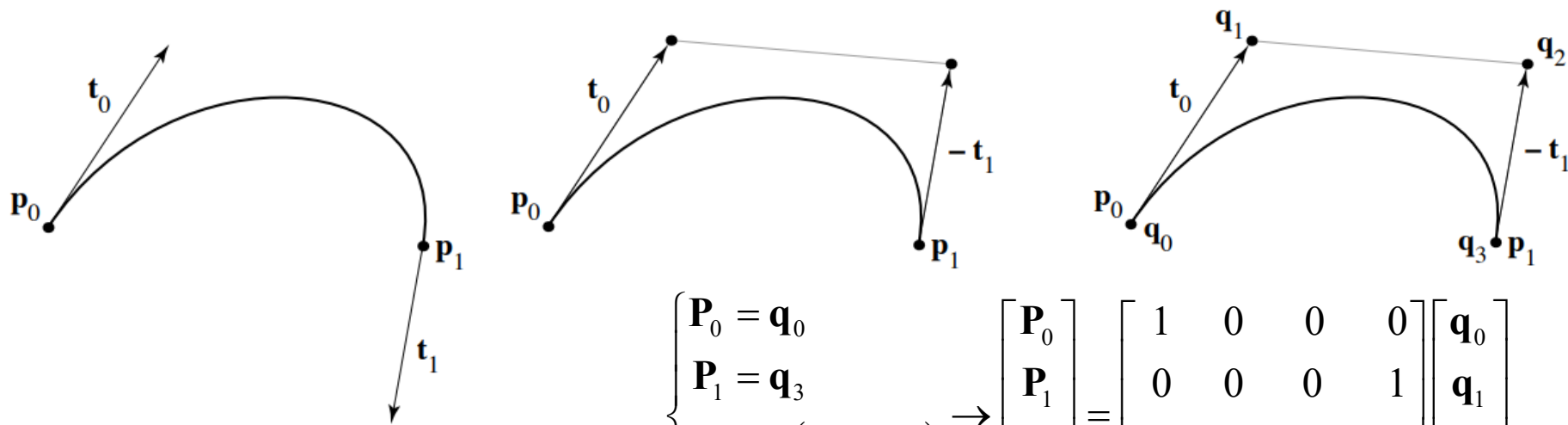
접선벡터의 영향

- cannot model data that have duplicate y-values
- curve shape is completely dependent on the data



Hermite to Bézier

- Mixture of points and vectors is awkward
- Specify tangents as differences of points



$$\begin{cases} \mathbf{P}_0 = \mathbf{q}_0 \\ \mathbf{P}_1 = \mathbf{q}_3 \\ \mathbf{t}_0 = 3(\mathbf{q}_1 - \mathbf{q}_0) \\ \mathbf{t}_1 = 3(\mathbf{q}_3 - \mathbf{q}_2) \end{cases} \rightarrow \begin{bmatrix} \mathbf{P}_0 \\ \mathbf{P}_1 \\ \mathbf{t}_0 \\ \mathbf{t}_1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ -3 & 3 & 0 & 0 \\ 0 & 0 & -3 & 3 \end{bmatrix} \begin{bmatrix} \mathbf{q}_0 \\ \mathbf{q}_1 \\ \mathbf{q}_2 \\ \mathbf{q}_3 \end{bmatrix}$$

$$\mathbf{r}(u) = \begin{bmatrix} 1 & u & u^2 & u^3 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ -3 & 3 & -2 & -1 \\ 2 & -2 & 1 & 1 \end{bmatrix} \begin{bmatrix} \mathbf{P}_0 \\ \mathbf{P}_1 \\ \mathbf{t}_0 \\ \mathbf{t}_1 \end{bmatrix} = \begin{bmatrix} 1 & u & u^2 & u^3 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ -3 & 3 & 0 & 0 \\ 3 & -6 & 3 & 0 \\ -1 & 3 & -3 & 1 \end{bmatrix} \begin{bmatrix} \mathbf{q}_0 \\ \mathbf{q}_1 \\ \mathbf{q}_2 \\ \mathbf{q}_3 \end{bmatrix}$$

$$\mathbf{r}(u) = \mathbf{a}_0 + \mathbf{a}_1 u + \mathbf{a}_2 u^2 + \mathbf{a}_3 u^3 \quad (0 \leq u \leq 1)$$

$$\mathbf{r}'(u) = \mathbf{a}_1 + 2\mathbf{a}_2 u + 3\mathbf{a}_3 u^2$$

$$\mathbf{r}(0) = \mathbf{a}_0 = \mathbf{P}_0$$

$$\mathbf{r}(1) = \sum \mathbf{a}_i = \mathbf{P}_1$$

$$\mathbf{r}'(0) = \mathbf{a}_1 = \mathbf{t}_0$$

$$\mathbf{r}'(1) = \sum i\mathbf{a}_i = \mathbf{t}_1$$

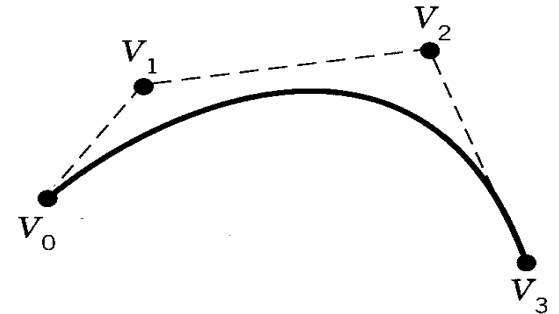
$$\left. \begin{array}{l} \mathbf{r}(0) = \mathbf{a}_0 = \mathbf{P}_0 \\ \mathbf{r}(1) = \sum \mathbf{a}_i = \mathbf{P}_1 \\ \mathbf{r}'(0) = \mathbf{a}_1 = \mathbf{t}_0 \\ \mathbf{r}'(1) = \sum i\mathbf{a}_i = \mathbf{t}_1 \end{array} \right\} \rightarrow \begin{bmatrix} \mathbf{P}_0 \\ \mathbf{P}_1 \\ \mathbf{t}_0 \\ \mathbf{t}_1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 \\ 0 & 1 & 0 & 0 \\ 0 & 1 & 2 & 3 \end{bmatrix} \begin{bmatrix} \mathbf{a}_0 \\ \mathbf{a}_1 \\ \mathbf{a}_2 \\ \mathbf{a}_3 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ -3 & 3 & 0 & 0 \\ 0 & 0 & -3 & 3 \end{bmatrix} \begin{bmatrix} \mathbf{q}_0 \\ \mathbf{q}_1 \\ \mathbf{q}_2 \\ \mathbf{q}_3 \end{bmatrix}$$

$$\begin{bmatrix} \mathbf{a}_0 \\ \mathbf{a}_1 \\ \mathbf{a}_2 \\ \mathbf{a}_3 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ -3 & 3 & 0 & 0 \\ 3 & -6 & 3 & 0 \\ -1 & 3 & -3 & 1 \end{bmatrix} \begin{bmatrix} \mathbf{q}_0 \\ \mathbf{q}_1 \\ \mathbf{q}_2 \\ \mathbf{q}_3 \end{bmatrix}$$

Bézier Curves

- History of Bézier Curve

- Pierre Bézier designed the Bézier curve
 - in the early 1960s
 - at Renault, French automobile company
 - UNISURF: surface modeler used by Renault since 1972 to design auto-bodies
- de Casteljaou at Citroen also designed the Bézier curve at the same time with Bézier's



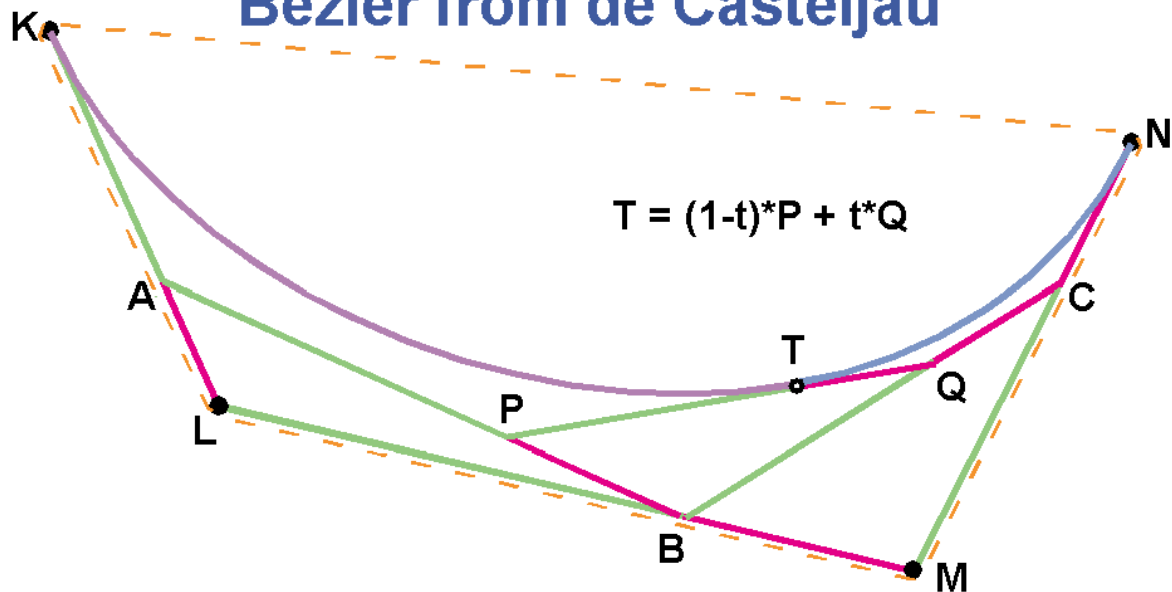
- Features of Bézier Curve

- Pass through the first and last control points
- Tangent to the lines joining the first two and last two control points
- No oscillation

de Casteljau's algorithm

- geometric evaluation scheme for Bézier
 - Recursive series of linear interpolations

Bezier from de Casteljau



$$P = (1-t)*A + t*B$$

$$Q = (1-t)*B + t*C$$

$$T = (1-t)(1-t)*A + (1-t)t*B + t(1-t)*B + t*t*C$$

$$A = (1-t)*K + t*L$$

$$B = (1-t)*L + t*M$$

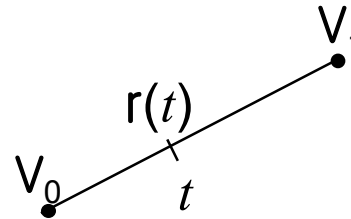
$$C = (1-t)*M + t*N$$

$$T = (1-t)^3*K + (1-t)^2*t*L + 2(1-t)^2*t*L + 2(1-t)t^2*M + (1-t)t^2*M + t^3*N$$

Bernstein Basis 에 의한 Bézier Curve

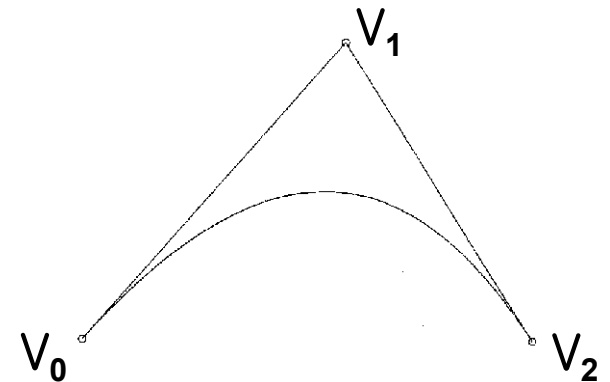
- 선형 Bézier 선

$$\mathbf{r}(t) = (1-t)\mathbf{V}_0 + t\mathbf{V}_1$$



- 2차 Bézier 곡선

$$\mathbf{r}(u) = (1-u)^2\mathbf{V}_0 + 2(1-u)u\mathbf{V}_1 + u^2\mathbf{V}_2$$



- n차 Bézier curve의 식

$$\mathbf{r}(u) = \sum_{i=0}^n B_i^n(u) \mathbf{V}_i, \quad 0 \leq u \leq 1$$

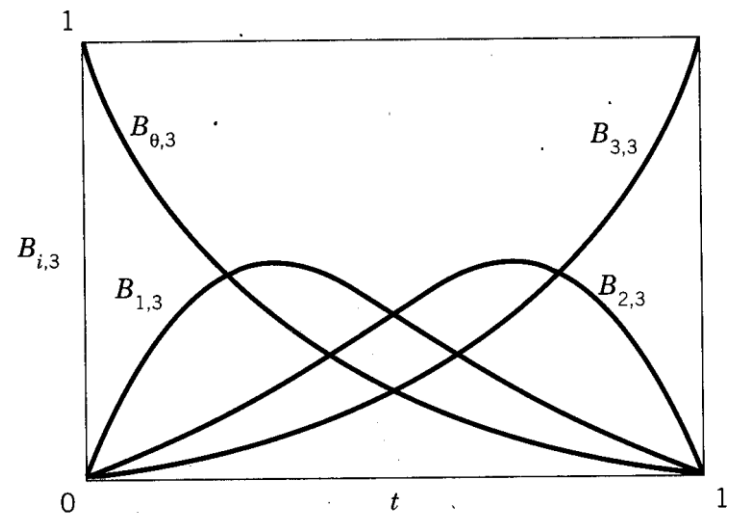
$$B_i^n(u) = \binom{n}{i} (1-u)^{n-i} u^i = \frac{n!}{i!(n-i)!} (1-u)^{n-i} u^i : \text{Bernstein basis function}$$

Bézier Curve의 일반식

$$B_{i,n}(u) = \binom{n}{i} (1-u)^{n-i} u^i \quad 0 < u \leq 1$$

- Blending function of Cubic Bézier Curve (degree 3)

$$\left\{ \begin{array}{l} B_{0,3}(u) = \frac{3!}{0!3!} (1-u)^3 u^0 = (1-u)^3 \\ B_{1,3}(u) = \frac{3!}{1!2!} (1-u)^2 u^1 = 3u(1-u)^2 \\ B_{2,3}(u) = \frac{3!}{2!1!} (1-u) u^2 = 3u^2(1-u) \\ B_{3,3}(u) = \frac{3!}{3!0!} (1-u)^0 u^3 = u^3 \end{array} \right.$$

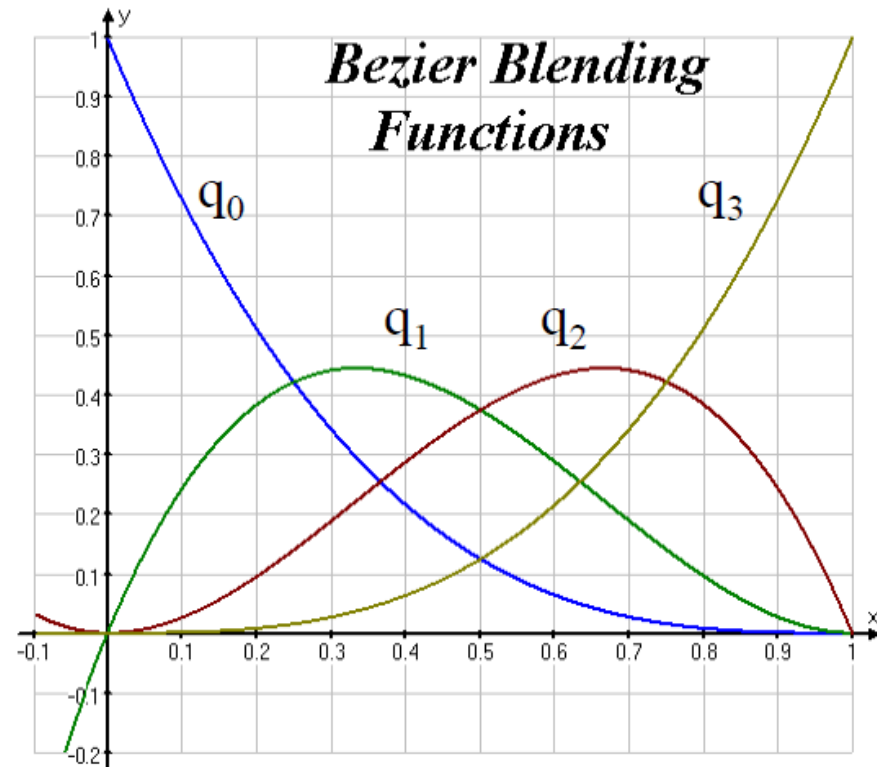
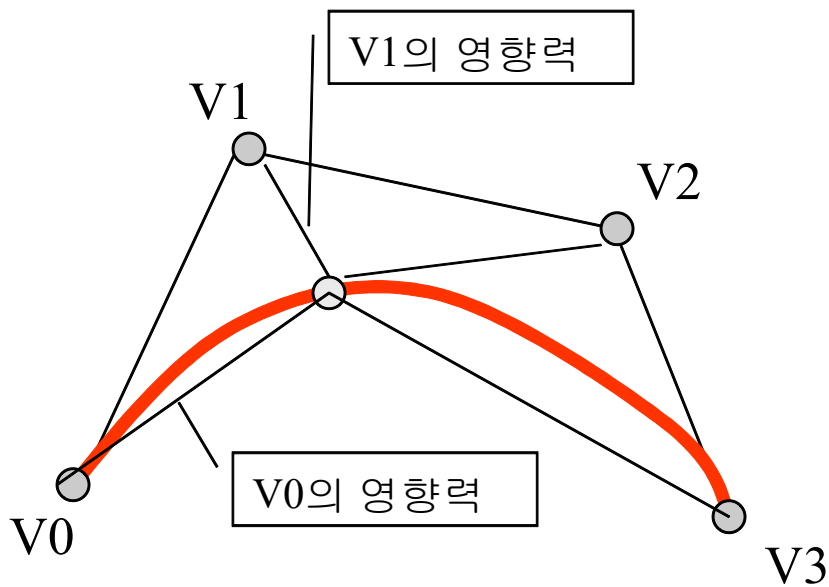


- Normalizing Property

$$(1-u)^3 + 3u(1-u)^2 + 3u^2(1-u) + u^3 = 1$$

Bézier Curve의 정의

- 4개의 조정점으로부터 영향력의 정도를 나타내는 블렌딩 함수 (Bernstein Blending function)를 이용하여 하나의 Bézier Curve를 정의



Cubic Bézier Curve

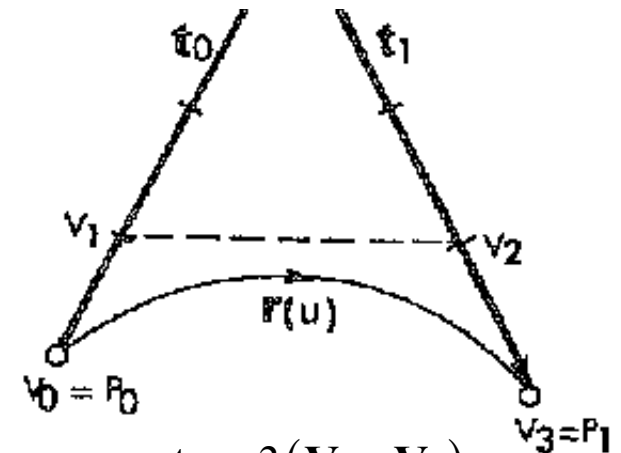
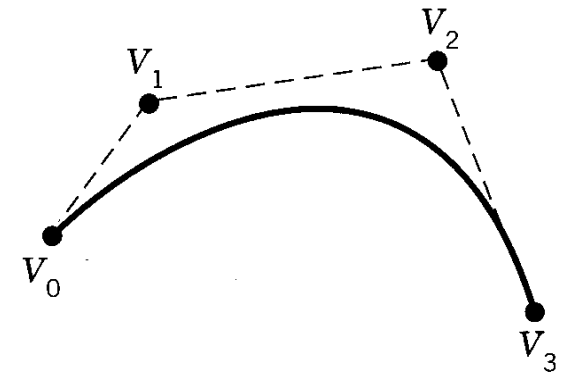
- 점 4개로 하나의 Bézier Curve를 정의

$$\begin{aligned} \mathbf{r}(u) &= (1-u)^3 \mathbf{V}_0 + 3u(1-u)^2 \mathbf{V}_1 + 3u^2(1-u) \mathbf{V}_2 + u^3 \mathbf{V}_3 \\ &= \mathbf{UMR} \quad (0 \leq u \leq 1) \end{aligned}$$

$$\mathbf{U} = [1 \quad u \quad u^2 \quad u^3]$$

$$\mathbf{M} = \begin{bmatrix} \phantom{\mathbf{V}_0} \\ \phantom{\mathbf{V}_1} \\ \phantom{\mathbf{V}_2} \\ \phantom{\mathbf{V}_3} \end{bmatrix}$$

$$\mathbf{R} = \begin{bmatrix} \mathbf{V}_0 \\ \mathbf{V}_1 \\ \mathbf{V}_2 \\ \mathbf{V}_3 \end{bmatrix} : \text{control points}$$



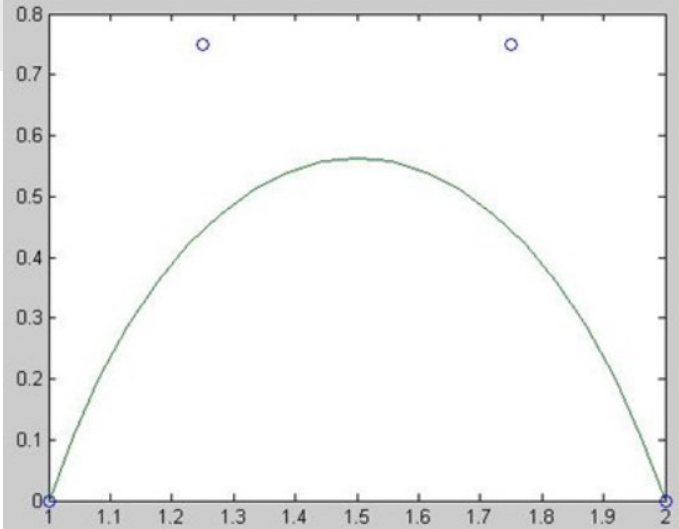
$$\mathbf{t}_0 = 3(\mathbf{V}_1 - \mathbf{V}_0)$$

$$\mathbf{t}_1 = 3(\mathbf{V}_2 - \mathbf{V}_3)$$

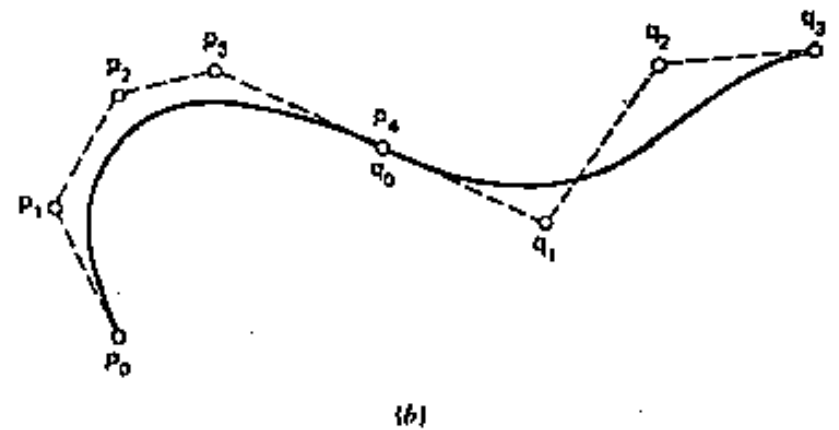
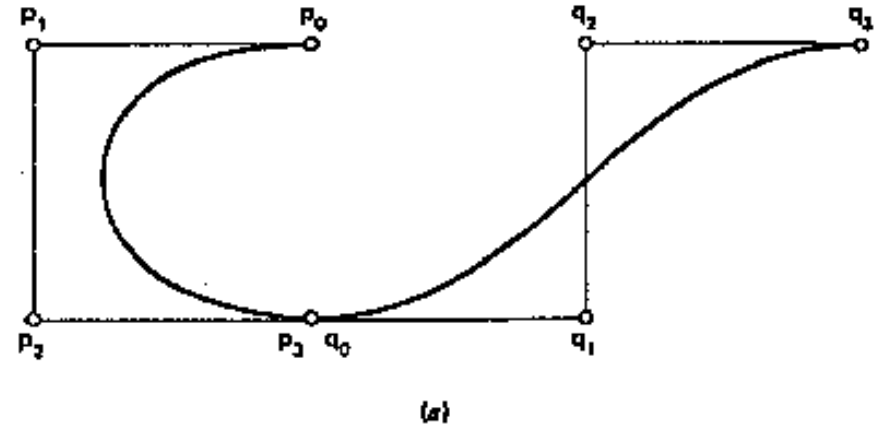
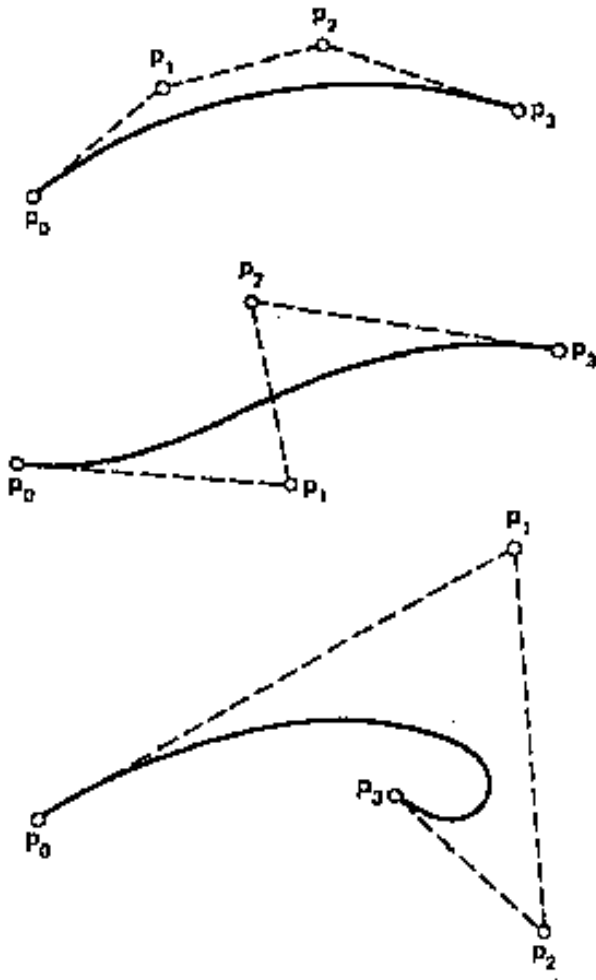
Matlab Function for Bézier Curves

```
function p = bezier(t, p0, p1, p2, p3 )  
    % bezier computes the bezier curve value p(t)  
    % defined by control points p0, p1, p2, p3  
    u=[t^3 t^2 t 1];  
    M = [-1, 3,-3, 1;  
          3,-6, 3, 0;  
          -3, 3, 0, 0;  
          1, 0, 0, 0]  
    C = [p0;p1;p2;p3]  
    p=u*M*C  
end
```

```
function plot_bezier(p0,p1,p2,p3)  
    %plot_bezier plots the bezier curve through the control  
    % points p0, p1, p2, p3  
    tvector = 0.0:0.05:1;  
    pts = zeros(length(tvector),2);  
    for i = 1:length(tvector)  
        pts(i,:) = bezier(tvector(i), p0, p1, p2, p3);  
    end  
    C = [p0;p1;p2;p3];  
    plot(C(:,1), C(:,2),'o',pts(:,1),pts(:,2),'-');  
end
```



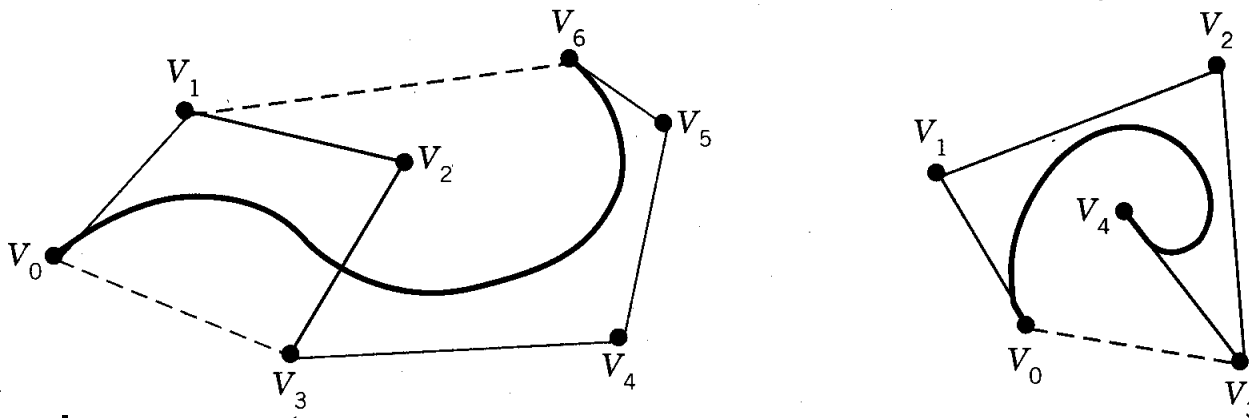
Bézier Curve의 예



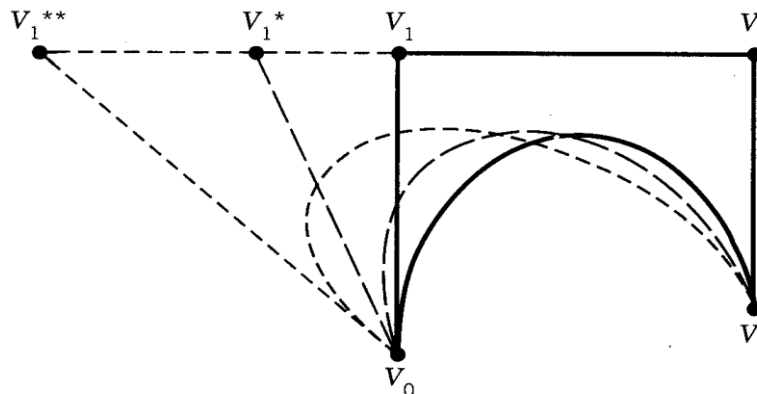
Composite Bézier curves

Bézier Curve의 성질 (1)

- Convex Hull Property
 - All points on curve inside convex hull of control points

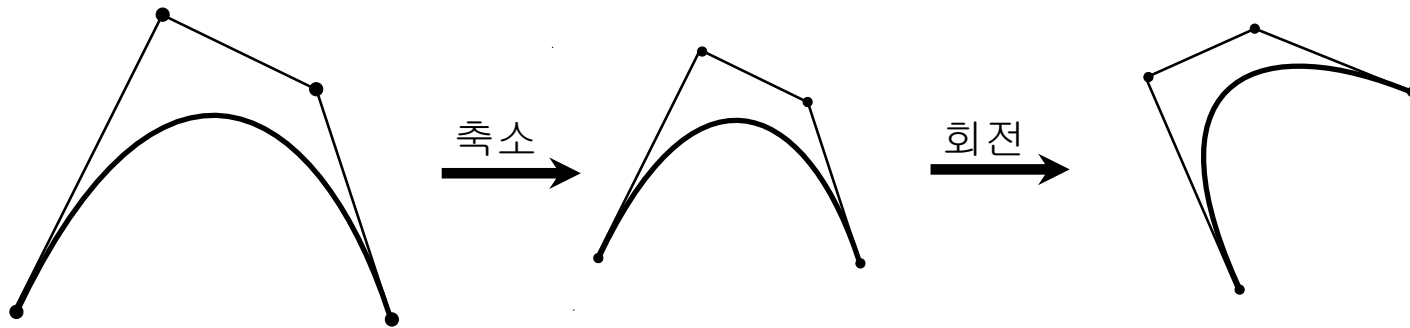


- Local support
 - Changing one control point has limited impact on entire curve

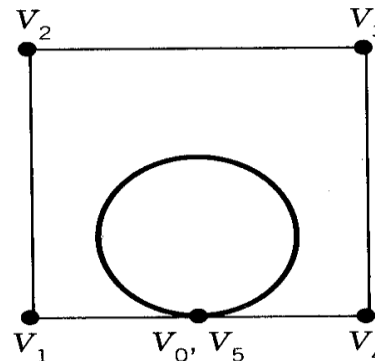


Bézier Curve의 성질 (2)

- Affine Invariance $r(u) = \sum_i v_i B_i(u) \Leftrightarrow Tr(u) = \sum_i (TV_i) B_i(u)$
 - Control point를 transformation하면 곡선도 같이 transform된다
 - Bézier basis invariant for affine transforms



- Interesting shapes by repeating control points
 - Three in a row: straight line
 - first = last: closed curve
 - Two in a row: higher weight

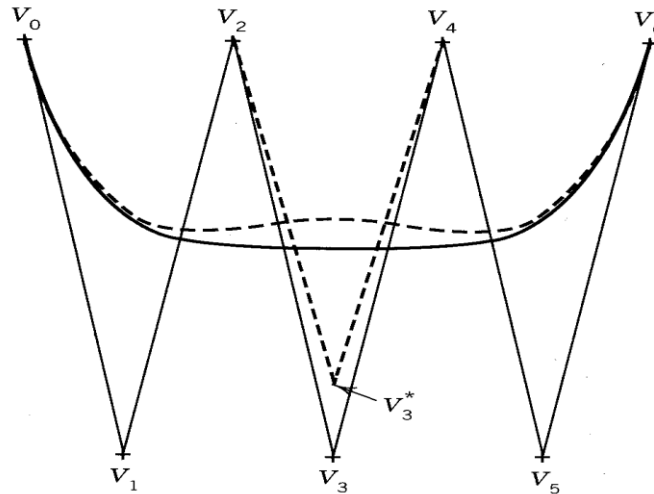


Bézier Curve의 성질 (3)

- 조정점의 개수와 곡선의 차수가 직결되어 모든 조정점이 곡선의 형상에 영향을 줌.

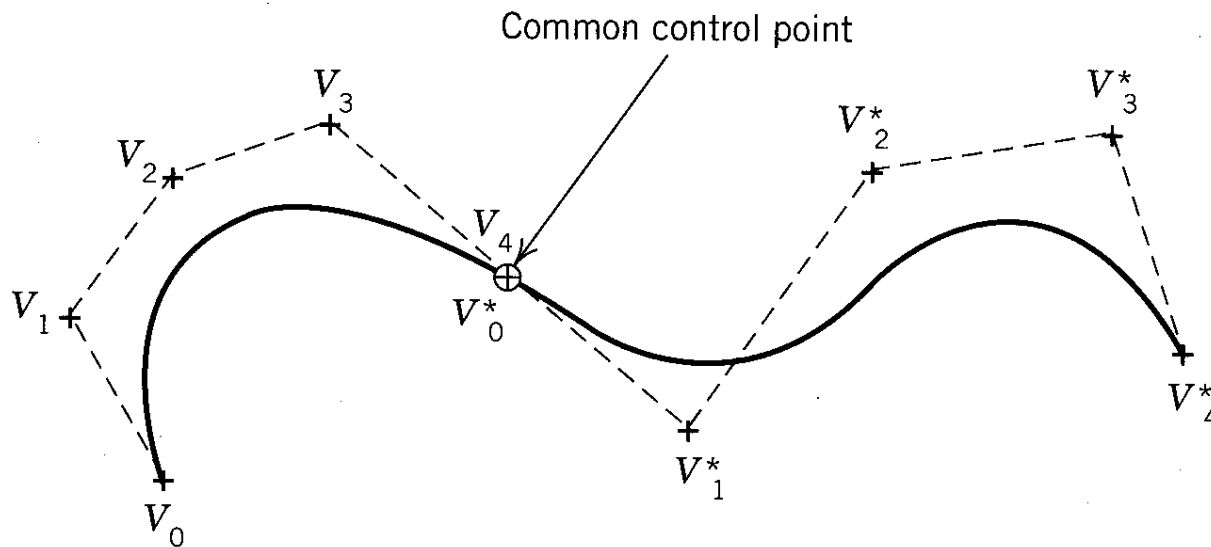
$$\mathbf{r}(u) = \sum_{i=0}^n B_i^n(u) \mathbf{V}_i, \quad 0 \leq u \leq 1, \quad B_i^n(u) = \binom{n}{i} (1-u)^{n-i} u^i$$

- 많은 조정점을 이용할 경우, 곡선식의 차수도 올라가게 되어 계산량이 증가되며 곡선이 진동하는 문제가 야기됨
- 모든 조정점이 곡선의 형상에 영향을 주기 때문에 곡선의 일부분을 변형시키면 나머지 부분도 예상치 못한 변화가 발생할 수 있음



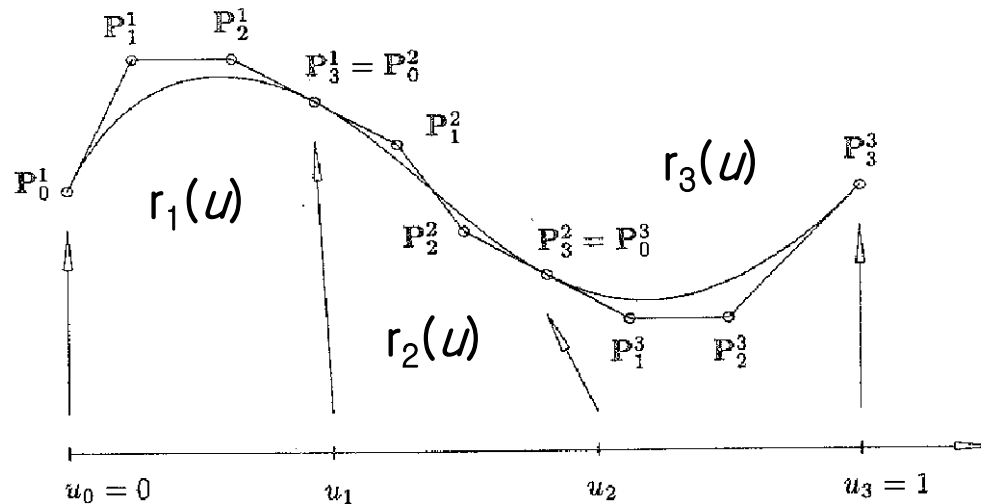
Composite Bézier Curve

- Piecewise Bézier curve in case of a large number of control points
 - C^1 continuity: three control points around the intersection are colinear



Composite Curve (복합 곡선)

- 복잡한 자유곡선을 표현하는 두가지 방법
 - 곡선의 차수 증가 → 원하지 않는 굴곡 발생
 - 여러 개의 곡선 결합으로 표현 → 대부분의 CAD/CAM system 이용
- Composite Curve: piecewisely defined continuous curve



$\{ r_1(u), r_2(u), r_3(u) \}$: composite curve
 $r_1(u), r_2(u), r_3(u)$: curve segment

Continuity (연속 조건)

- Parametric Continuity

- differentiability of the parametric representation

C^0 연속

$$\mathbf{r}_1(u_1) = \mathbf{r}_2(u_1)$$

C^1 연속

$$\frac{d}{du} \mathbf{r}_1(u_1) = \frac{d}{du} \mathbf{r}_2(u_1)$$

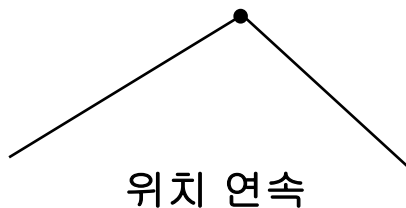
C^2 연속

$$\frac{d^2}{du^2} \mathbf{r}_1(u_1) = \frac{d^2}{du^2} \mathbf{r}_2(u_1)$$

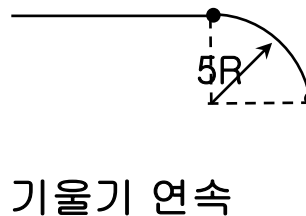
- Geometric Continuity

- smoothness of the resulting displayed shape

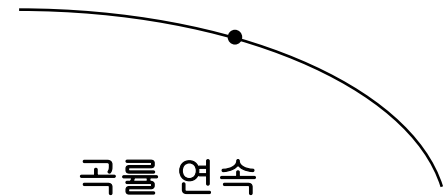
G^0 연속 (C^0 연속)



G^1 연속 (방향만 동일)



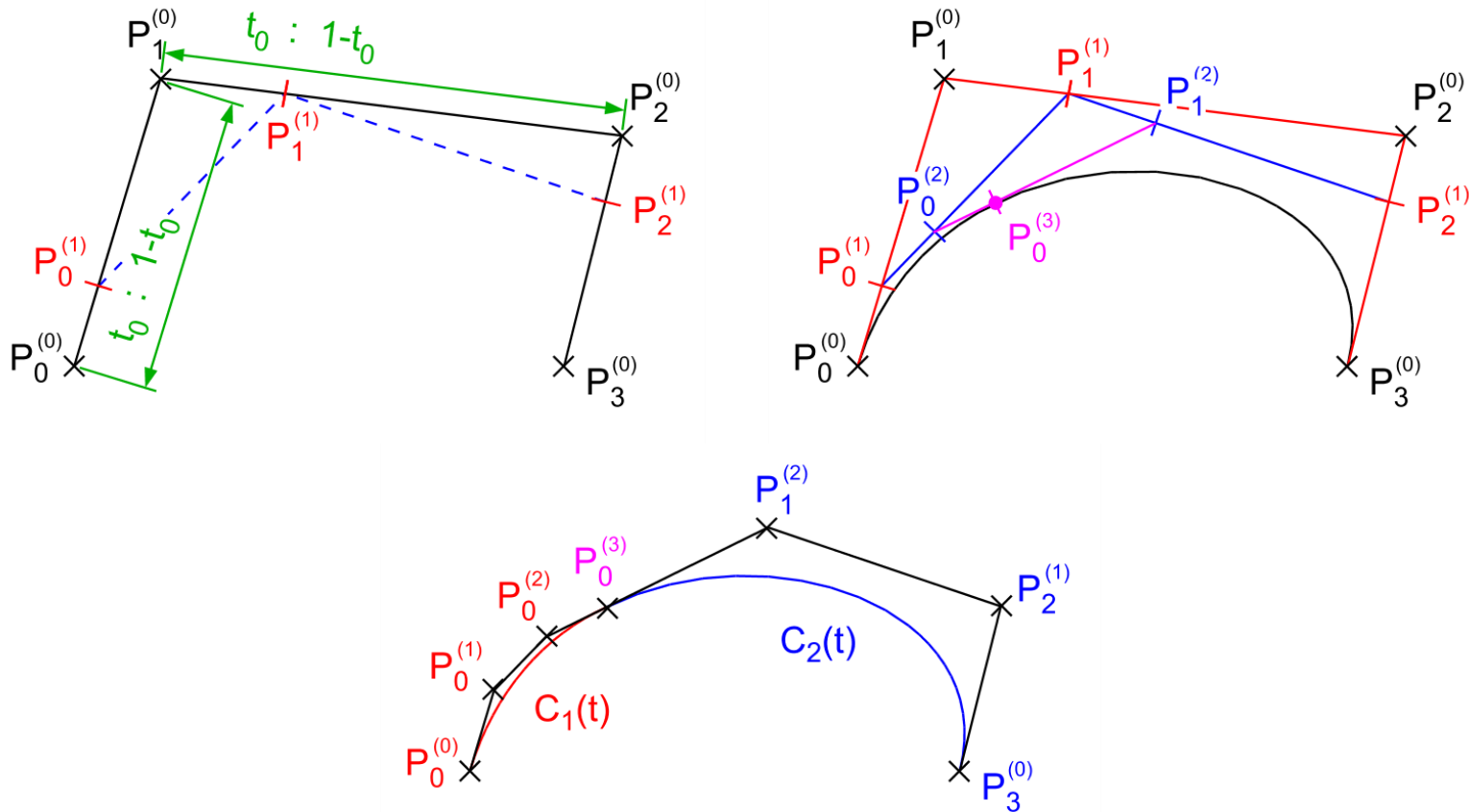
G^2 연속



- G^x 연속은 curve의 재매개변수화에 의해 C^x 연속으로 바꿀 수 있다.

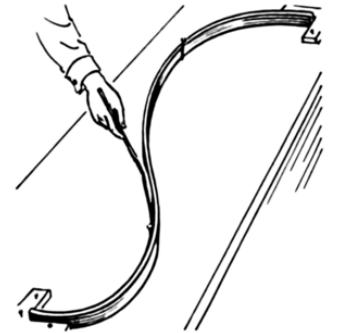
Bézier subdivision: de Casteljau's algorithm

- split a single Bézier curve into two Bézier curves at an arbitrary parameter value

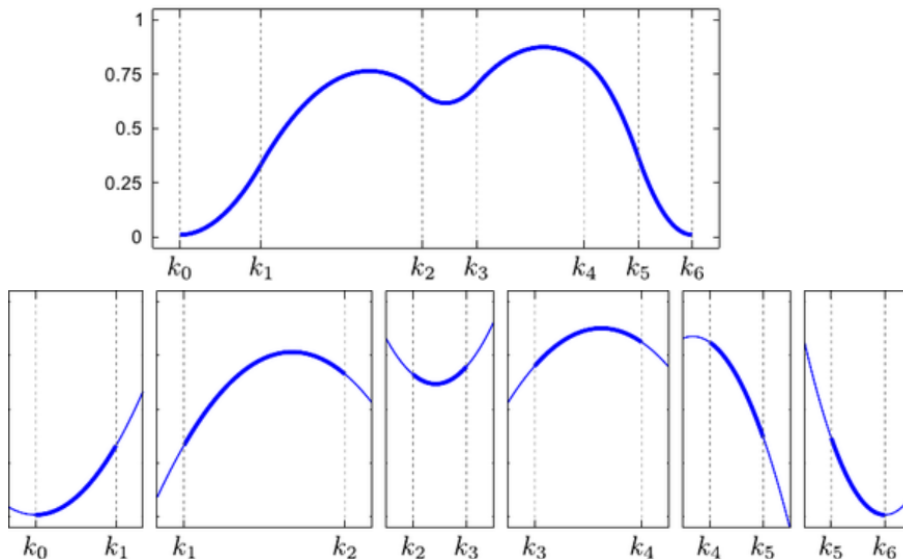


Natural Cubic Splines (1)

- Draw a “smooth” line through several points
- Given $n+1$ points
 - Generate a curve with n segments
 - Curve passes through points
 - Curve is C^2 continuous
- Use cubics because lower order is better...



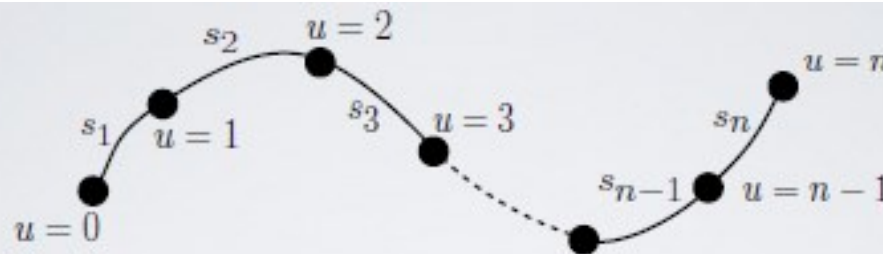
An historical spline
(credits: Pearson Scott Foresman)



$$x(u) = \begin{cases} s_1(u) & \text{if } 0 \leq u < 1 \\ s_2(u-1) & \text{if } 1 \leq u < 2 \\ s_3(u-2) & \text{if } 2 \leq u < 3 \\ \vdots & \vdots \\ s_n(u-(n-1)) & \text{if } n-1 \leq u \leq n \end{cases}$$

Natural Cubic Splines (2)

- Interpolate data points
- No convex hull property
- Non-local support



$$s_i(0) = p_{i-1} \quad i = 1 \dots n \quad \leftarrow n \text{ constraints}$$

$$s_i(1) = p_i \quad i = 1 \dots n \quad \leftarrow n \text{ constraints}$$

$$s'_i(1) = s'_{i+1}(0) \quad i = 1 \dots n-1 \quad \leftarrow n-1 \text{ constraints}$$

$$s''_i(1) = s''_{i+1}(0) \quad i = 1 \dots n-1 \quad \leftarrow n-1 \text{ constraints}$$

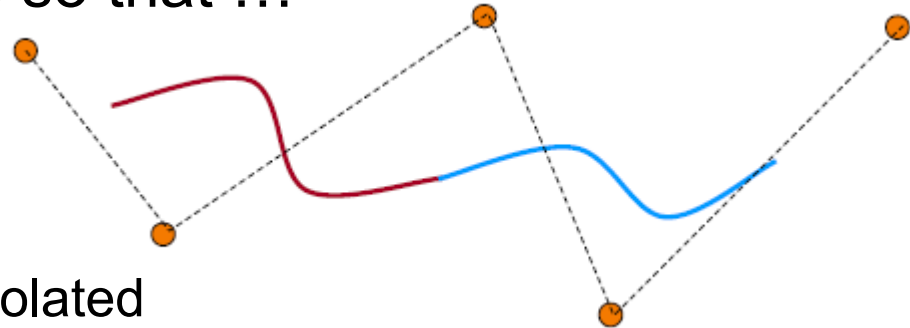
$$s''_1(0) = s''_n(1) = 0 \quad \leftarrow 2 \text{ constraints}$$

Total $4n$ constraints

Uniform Cubic B-splines (1)

- Choose blending functions so that ...

- Cubic polynomials
- C^2 continuity
- Local control
- Points not necessarily interpolated



- Derivation

- Three continuity conditions for each joint J_i ...
 - Position, derivatives and second derivatives of two curves are equal at J_i
- Also, local control implies ...
 - Each joint is affected by small set of (4) points

Uniform Cubic B-splines (2)

$$Q_1(u) = b_0(u)V_0 + b_1(u)V_1 + b_2(u)V_2 + b_3(u)V_3$$

$$Q_2(u) = b_0(u)V_1 + b_1(u)V_2 + b_2(u)V_3 + b_3(u)V_4$$

(15 continuity constraints)

$$Q_1(1) = Q_2(0) \quad Q'_1(1) = Q'_2(0) \quad Q''_1(1) = Q''_2(0)$$

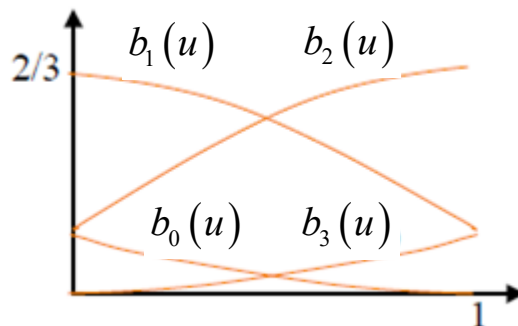
$$\rightarrow \begin{cases} b_0(1) = 0 \\ b_1(1) = b_0(0) \\ b_2(1) = b_1(0) \\ b_3(1) = b_2(0) \\ 0 = b_3(0) \end{cases} \rightarrow \begin{cases} b'_0(1) = 0 \\ b'_1(1) = b'_0(0) \\ b'_2(1) = b'_1(0) \\ b'_3(1) = b'_2(0) \\ 0 = b'_3(0) \end{cases} \rightarrow \begin{cases} b''_0(1) = 0 \\ b''_1(1) = b''_0(0) \\ b''_2(1) = b''_1(0) \\ b''_3(1) = b''_2(0) \\ 0 = b''_3(0) \end{cases}$$

(1 normality constraint)

$$b_0(0) + b_1(0) + b_2(0) + b_3(0) = 1$$

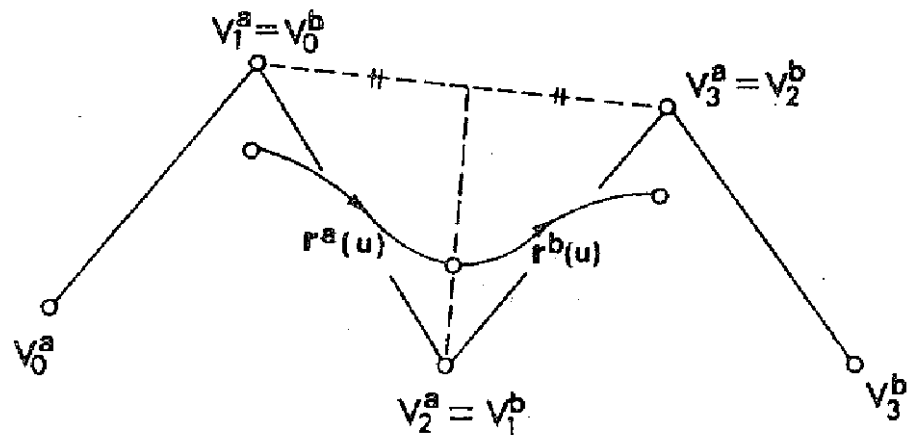
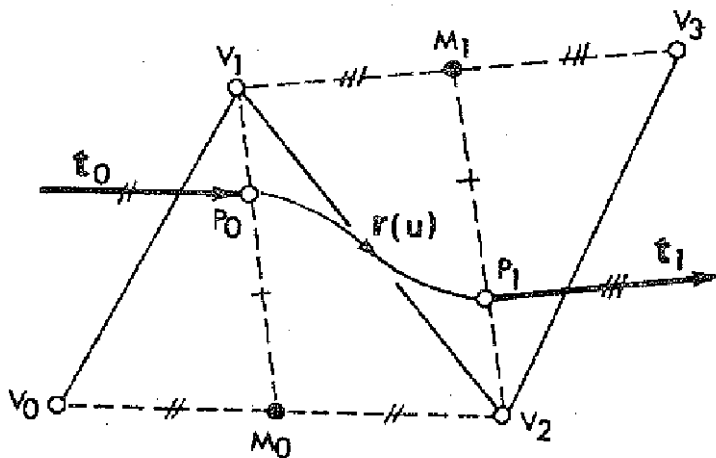
$$\left. \begin{aligned} b_0(u) &= a_{00} + a_{01}u + a_{02}u^2 + a_{03}u^3 \\ b_1(u) &= a_{10} + a_{11}u + a_{12}u^2 + a_{13}u^3 \\ b_2(u) &= a_{20} + a_{21}u + a_{22}u^2 + a_{23}u^3 \\ b_3(u) &= a_{30} + a_{31}u + a_{32}u^2 + a_{33}u^3 \end{aligned} \right\}$$

$$\rightarrow \begin{cases} b_0(u) = \frac{1}{6} - \frac{1}{2}u + \frac{1}{2}u^2 - \frac{1}{6}u^3 \\ b_1(u) = \frac{2}{3} + 0u - u^2 + \frac{1}{2}u^3 \\ b_2(u) = \frac{1}{6} + \frac{1}{2}u + \frac{1}{2}u^2 - \frac{1}{6}u^3 \\ b_3(u) = 0 + 0u + 0u^2 + \frac{1}{6}u^3 \end{cases}$$



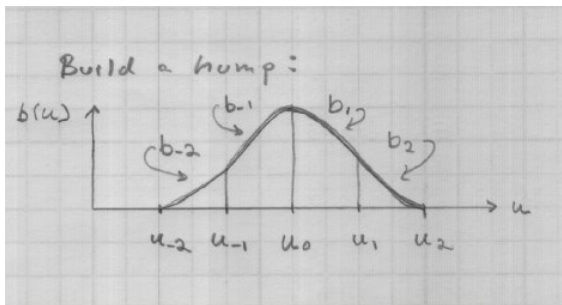
Uniform Cubic B-Spline Curve (3)

$$\begin{aligned}
 \mathbf{r}(u) &= b_0(u) \mathbf{V}_0 + b_1(u) \mathbf{V}_1 + b_2(u) \mathbf{V}_2 + b_3(u) \mathbf{V}_3 \\
 &= \left(\frac{1}{6} - \frac{1}{2}u + \frac{1}{2}u^2 - \frac{1}{6}u^3 \right) \mathbf{V}_0 + \left(\frac{2}{3} + 0u - u^2 + \frac{1}{2}u^3 \right) \mathbf{V}_1 \\
 &\quad + \left(\frac{1}{6} + \frac{1}{2}u + \frac{1}{2}u^2 - \frac{1}{6}u^3 \right) \mathbf{V}_2 + \left(0 + 0u + 0u^2 + \frac{1}{6}u^3 \right) \mathbf{V}_3 \\
 &= \begin{bmatrix} 1 & u & u^2 & u^3 \end{bmatrix} \frac{1}{6} \begin{bmatrix} 1 & 4 & 1 & 0 \\ -3 & 0 & 3 & 0 \\ 3 & -6 & 3 & 0 \\ -1 & 3 & -1 & 1 \end{bmatrix} \begin{bmatrix} \mathbf{V}_0 \\ \mathbf{V}_1 \\ \mathbf{V}_2 \\ \mathbf{V}_3 \end{bmatrix} = \mathbf{UNR} \quad (0 \leq u \leq 1)
 \end{aligned}$$



B-splines (1)

- Goal: C^2 cubic curves with local support
 - Give up interpolation
 - Get convex hull property
- Build basis by designing “hump” functions



$$b(u) = \begin{cases} b_{-2}(u) & \text{if } u_{-2} \leq u < u_{-1} \\ b_{-1}(u) & \text{if } u_{-1} \leq u < u_0 \\ b_{+1}(u) & \text{if } u_0 \leq u < u_{+1} \\ b_{+2}(u) & \text{if } u_{+1} \leq u \leq u_{+2} \end{cases}$$

$$\begin{aligned} b''_{-2}(u_{-2}) = b'_{-2}(u_{-2}) = b_{-2}(u_{-2}) = 0 & \leftarrow 3 \text{ constraints} \\ b''_{+2}(u_{+2}) = b'_{+2}(u_{+2}) = b_{+2}(u_{+2}) = 0 & \leftarrow 3 \text{ constraints} \end{aligned}$$

$$\begin{aligned} b_{-2}(u_{-1}) &= b_{-1}(u_{-1}) \\ b_{-1}(u_0) &= b_{+1}(u_0) \\ b_{+1}(u_{+1}) &= b_{+2}(u_{+1}) \end{aligned} \leftarrow \begin{cases} \text{Repeat for } b' \text{ and } b'' \\ 3 \times 3 = 9 \text{ constraints} \end{cases}$$

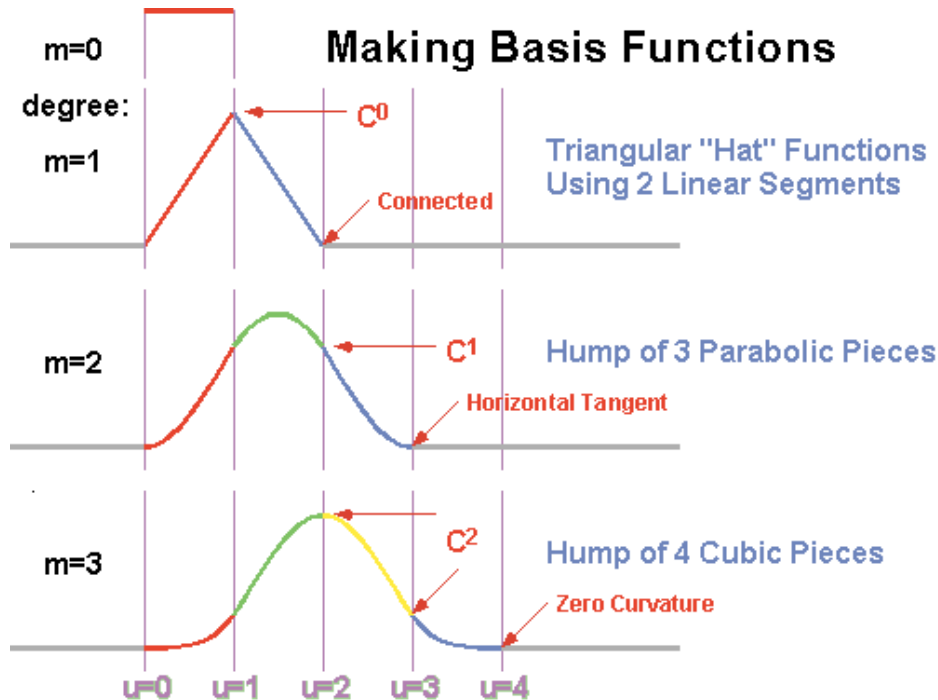
$$b_{-2}(u_{-2}) + b_{-1}(u_{-1}) + b_{+1}(u_0) + b_{+2}(u_{+1}) = 1 \leftarrow 1 \text{ constraint (convex hull)}$$

Total 16 constraints

B-splines (2)

- Build a curve w/ overlapping bumps
- C^2 Continuity
 - Inside bumps, Bumps “fade out”
- Notation
 - The basis functions are the $b_i(u)$
 - Hump functions are the concatenated function
 - The u_i 's are the knot locations
 - The weights on the hump/basis functions are control points
 - Let hump centered on u_i be $N_{i,k}(u)$: order k

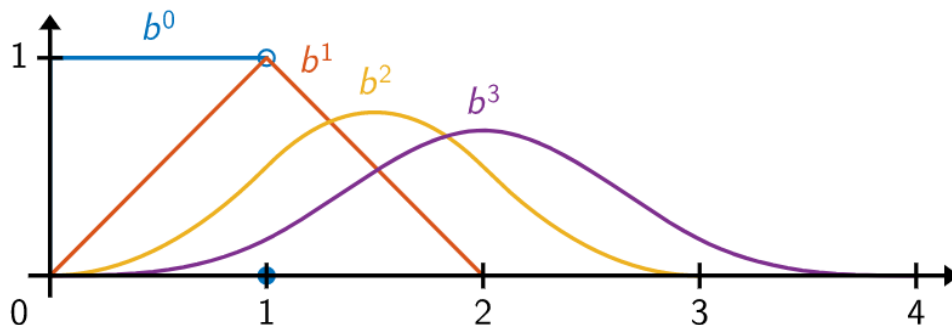
B-spline Basis Functions (1)



$$N_{i,1}(u) = \begin{cases} 1 & \text{if } u_i \leq u < u_{i+1} \\ 0 & \text{else} \end{cases}$$

$$N_{i,2}(u) = \frac{u-u_i}{u_{i+1}-u_i} N_{i,1}(u) + \frac{u_{i+2}-u}{u_{i+2}-u_{i+1}} N_{i+1,1}(u)$$

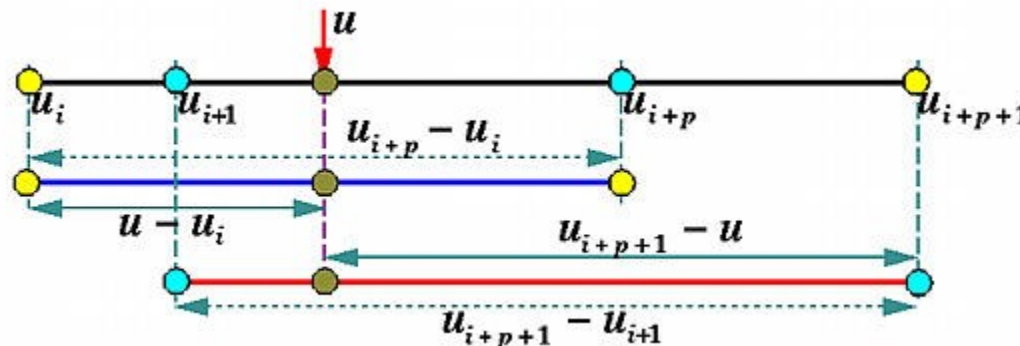
$$= uN_{i,1}(u) + (2-u)N_{i+1,1}(u) = \begin{cases} u \\ 2-u \end{cases}$$



B-spline Basis Functions (2)

- Basis function $N_{i,k}(u)$ is non-zero on
 - $[u_i, u_{i+k+1})$
 - $(k+1)$ knot span: $[u_i, u_{i+1}), [u_{i+1}, u_{i+2}), \dots, [u_{i+k}, u_{i+k+1})$
- On any knot span $[u_i, u_{i+1})$, at most k basis functions are non-zero
- Meaning of coefficients?

$$N_{i,k}(u) = \frac{u-u_i}{u_{i+k}-u_i} N_{i,k-1}(u) + \frac{u_{i+k+1}-u}{u_{i+k+1}-u_{i+1}} N_{i+1,k-1}(u)$$



B-spline Curve의 정의

$$\mathbf{P}(u) = \sum_{i=0}^n \mathbf{P}_i N_{i,k}(u) \quad (t_0 \leq u \leq t_{n+k})$$

$\{\mathbf{P}_0, \mathbf{P}_1, \dots, \mathbf{P}_n\}$: control point

$N_{i,k}(u)$: blending function of degree $(k-1)$

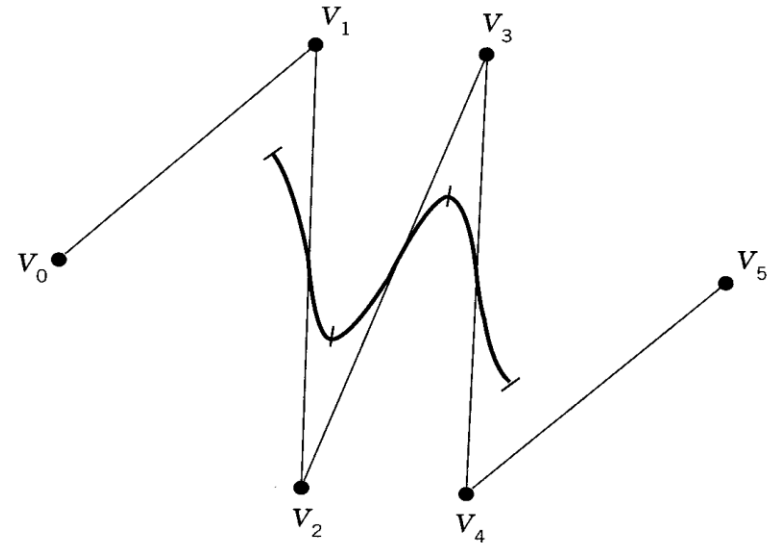
$$N_{i,k}(u) = \frac{(u - t_i)}{t_{i+k-1} - t_i} N_{i,k-1}(u) + \frac{(t_{i+k} - u)}{t_{i+k} - t_{i+1}} N_{i+1,k-1}(u)$$

Cox-de Boor (blending) function

$$N_{i,1}(u) = \begin{cases} 1 & t_i \leq u \leq t_{i+1} \\ 0 & \text{otherwise} \end{cases}$$

t_i : u 의 범위 내에 존재하는 매듭값 (knot value)

k : order (degree = $k-1$)



B-spline of order k in the i -th span is the weighted average of the B-splines of order $(k-1)$ in the i -th and $(i+1)$ -th spans

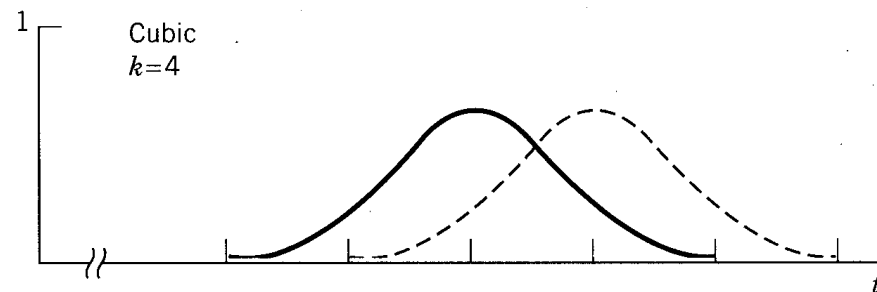
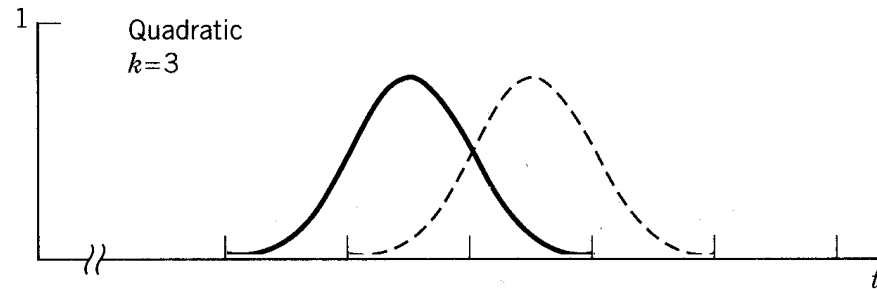
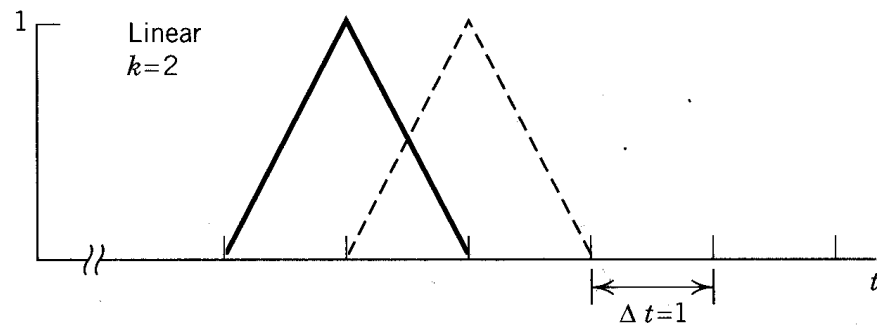
- Convex hull property
- Normalizing property

$$\sum_{i=0}^n N_{i,k}(u) = 1$$

B-Spline Curve의 특성

- 조정점의 개수와 다항식의 차수가 서로 독립적이다.
 - 설계자가 원하는 차수를 직접 정할 수 있음.
 - Bezier curve에서는 조정점의 개수 = 차수 + 1
- 국부적인 형상 조정이 가능하다.
 - 모든 블렌딩 함수는 매개변수 u 의 전체 범위 중 각각 서로 다른 일정 범위에서만 값을 갖도록 함.
 - Bezier curve에서는 블렌딩 함수가 u 의 전체범위에서 값을 가짐 $\rightarrow$ 형상이 전체적으로 바뀜
- Degree가 3차 이상(order는 4차이상)이면 2차 미분 연속이 보장됨

B-spline Curve Blending Functions



Knot Vector

- Relationship of parameters

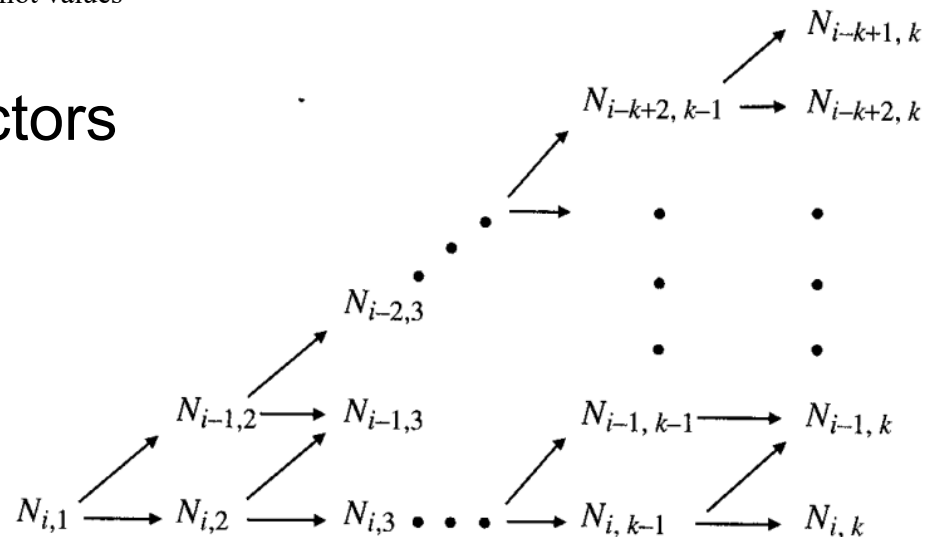
$$\underbrace{(m+1)}_{\text{\# of knots}} = \underbrace{(n+1)}_{\text{\# of control points}} + \underbrace{k}_{\text{order of curve}}$$

$$m = n + k$$

$$\underbrace{N_{0,k}(u), \dots, N_{n,k}(u)}_{(n+1) \text{ blending functions}} \leftarrow \underbrace{\{t_0, \dots, t_{n+k}\}}_{(n+k+1) \text{ knot values}}$$

- Classification of knot vectors

- Uniform / periodic
- Nonperiodic
- Nonuniform



Uniform/Periodic (1)

- Uniform knot vector has equispaced t_i values

- Let $(t_i - t_{i-1}) = a$

- $[0 \ 1 \ 2 \ 3 \ 4]$ with $a = 1$

- $[-0.5 \ 0.0 \ 0.5 \ 1.0 \ 1.5]$ with $a = 0.5$

$$\begin{cases} n = 5 \\ \# \text{ of control points} = n + 1 = 6 \end{cases}$$

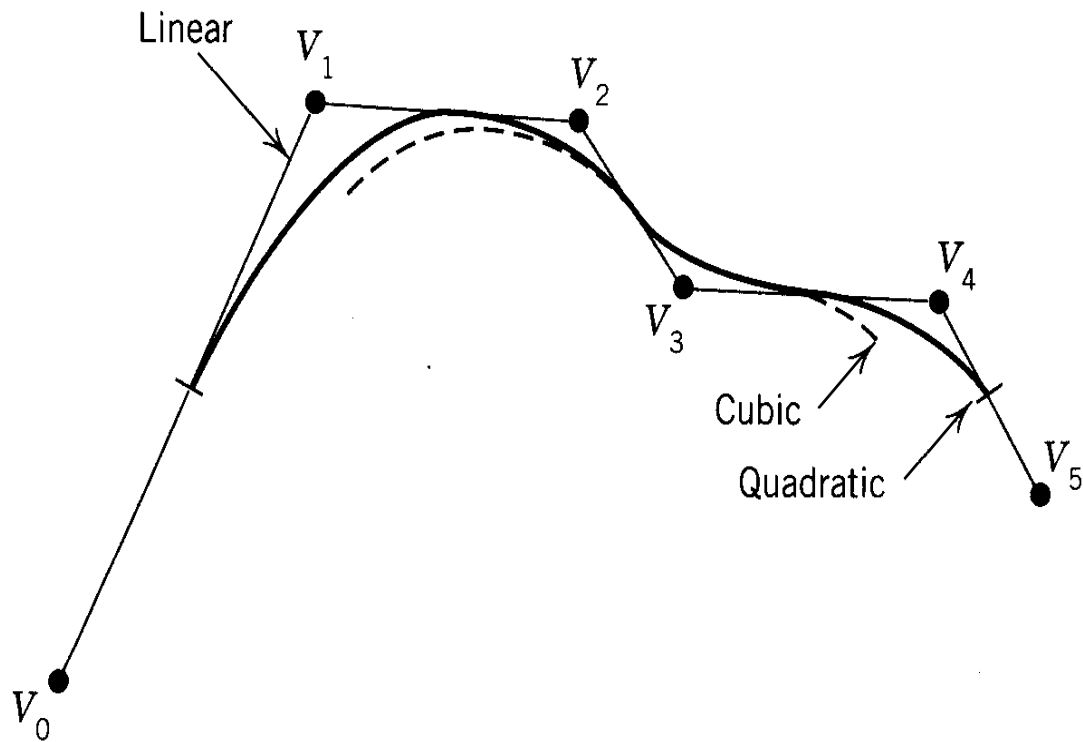
Degree ($k - 1$)	Order (k)	Knot Vector ($m = n + k$)	Parameter Range ($k - 1 \leq t \leq (n + 1)$)
1	2	$[0 \ 1 \ 2 \ 3 \ 4 \ 5 \ 6 \ 7]$	$1 \leq t \leq 6$
2	3	$[0 \ 1 \ 2 \ 3 \ 4 \ 5 \ 6 \ 7 \ 8]$	$2 \leq t \leq 6$
3	4	$[0 \ 1 \ 2 \ 3 \ 4 \ 5 \ 6 \ 7 \ 8 \ 9]$	$3 \leq t \leq 6$

- Normalized in the range of $[0 \text{ to } 1]$

- $[0 \ \frac{1}{4} \ \frac{1}{2} \ \frac{3}{4} \ 1]$

Uniform/Periodic (2)

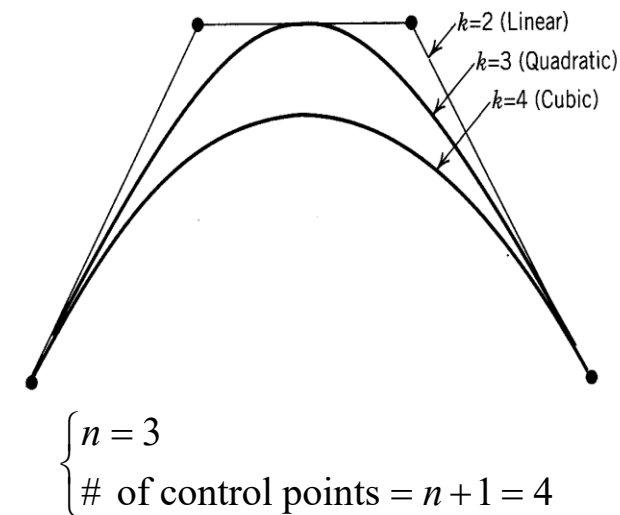
- Uniform B-splines of various degrees



Nonperiodic (1)

- Nonperiodic or Open Knot Vector
 - has repeated knot values at the ends with multiplicity equal to the order of the function k and internal knots equally spaced

Order (k)	No. of knots ($m = n + k$)	Nonperiodic knot vector
2	6	$[\underbrace{0 \ 0}_k \ 1 \ 2 \ \underbrace{3 \ 3}_k]$
3	7	$[\underbrace{0 \ 0 \ 0}_k \ 1 \ \underbrace{2 \ 2 \ 2}_k]$
4	8	$[\underbrace{0 \ 0 \ 0 \ 0}_k \ \underbrace{1 \ 1 \ 1 \ 1}_k]$



- General expression

$$\underbrace{t_0, \dots, t_{n+k}}_{(n+k+1) \text{ knot values}} \rightarrow \begin{cases} \text{periodic: } t_i = i - k, & 0 \leq i \leq n+k \\ \text{nonperiodic: } t_i = \begin{cases} 0 & 0 \leq i < k \\ i - k + 1 & k \leq i \leq n \\ n - k + 2 & n < i \leq n+k \end{cases} \end{cases}$$

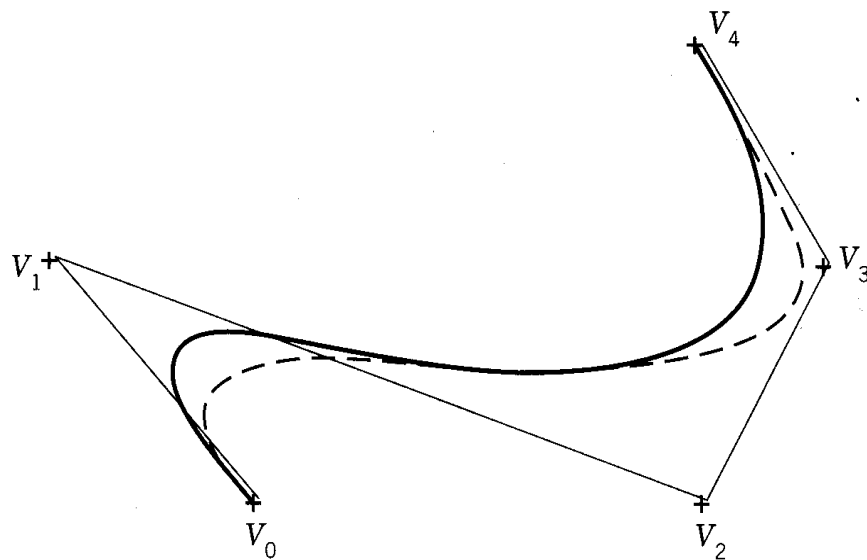
Nonperiodic (2)

- No loss of parameter range
 - Curve interpolates the first and last control points
 - $0 \leq u \leq n-k+2$
- Bézier curve: special case of nonperiodic B-spline
 - If no. of control points $(n+1) = \text{order } (k)$ and a nonperiodic uniform knot vector is used
 - Cubic B-spline with 4 control points and a knot vector $[0 \ 0 \ 0 \ 0 \ 1 \ 1 \ 1 \ 1]$ $\rightarrow$ Cubic Bézier curve

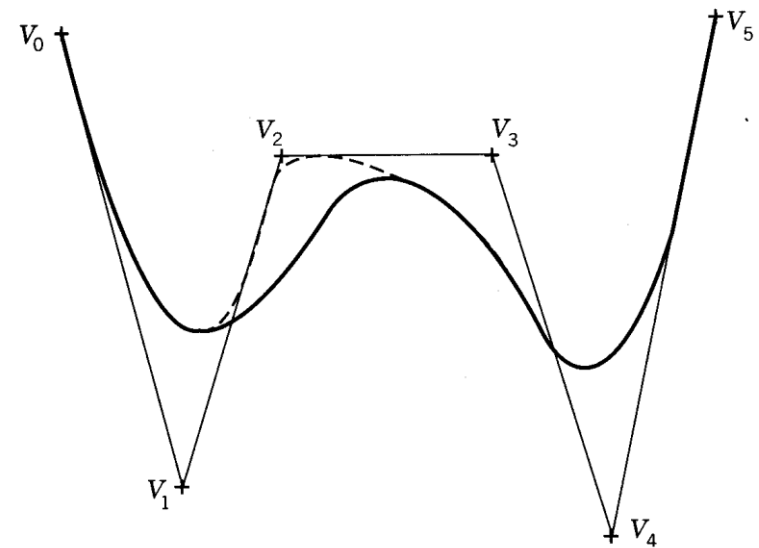
$$\underbrace{[0 \quad 0 \quad \dots \quad 0]}_k \quad \underbrace{1 \quad 1 \quad \dots \quad 1]}_k$$

Nonperiodic (3)

- Multiple interior knot values or unequal spacing
 - $[0 \ 1 \ 2 \ 3 \ 3 \ 4]$
 - $[0.0 \ 0.20 \ 0.55 \ 0.75 \ 1.0]$
- Effect of multiplicity of control points
 - Generate a span of zero length
 - C^{k-m-2} continuity at t_i
 - $m (\leq k-2)$ is the multiplicity of interior knot value



— $k = 4$, KNOT $[0 \ 0 \ 0 \ 0 \ 3 \ 7 \ 7 \ 7 \ 7]$
 - - - $k = 4$, KNOT $[0 \ 0 \ 0 \ 0 \ 8 \ 10 \ 10 \ 10 \ 10]$



— No multiplicity
 - - - Multiplicity 2 at V_2

B-spline Curve의 예 (1)

P_0, P_1, P_2 의 조정점을 갖고 order (k)가 3인 비주기적 B-spline 곡선
비 주기 매듭값 t_i 는 다음과 같음.

$$t_0 = 0, t_1 = 0, t_2 = 0, t_3 = 1, t_4 = 1, t_5 = 1$$

k=1에 해당되는
블렌딩 함수 $N_{i,1}$ 을 유도.

$$N_{0,1} = \begin{cases} 1 & t_0 \leq u < t_1 \\ 0 & \text{otherwise} \end{cases}$$

$$N_{1,1} = \begin{cases} 1 & t_1 \leq u < t_2 \\ 0 & \text{otherwise} \end{cases}$$

$$N_{2,1} = \begin{cases} 1 & t_2 \leq u < t_3 \\ 0 & \text{otherwise} \end{cases}$$

$$N_{3,1} = \begin{cases} 1 & t_3 \leq u < t_4 \\ 0 & \text{otherwise} \end{cases}$$

$$N_{4,1} = \begin{cases} 1 & t_4 \leq u < t_5 \\ 0 & \text{otherwise} \end{cases}$$

k=2에 해당되는
블렌딩 함수 $N_{i,2}$ 을 유도.

$$N_{0,2} = \frac{(u-t_0)N_{0,1}}{t_1-t_0} + \frac{(t_2-u)N_{1,1}}{t_2-t_1} = \frac{uN_{0,1}}{0} + \frac{(-u)N_{1,1}}{0} = 0$$

$$N_{1,2} = \frac{(u-t_1)N_{1,1}}{t_2-t_1} + \frac{(t_3-u)N_{2,1}}{t_3-t_2} = \frac{uN_{1,1}}{0} + \frac{(1-u)N_{2,1}}{1} = (1-u)$$

$$N_{2,2} = \frac{(u-t_2)N_{2,1}}{t_3-t_2} + \frac{(t_4-u)N_{3,1}}{t_4-t_3} = \frac{uN_{2,1}}{1} + \frac{(1-u)N_{3,1}}{0} = u$$

$$N_{3,2} = \frac{(u-t_3)N_{3,1}}{t_4-t_3} + \frac{(t_5-u)N_{4,1}}{t_5-t_4} = \frac{(u-1)N_{3,1}}{0} + \frac{(1-u)N_{4,1}}{0} = 0$$

B-spline Curve의 예 (2)

k=3에 해당되는 블렌딩 함수 $N_{i,3}$ 을 유도.

$$N_{0,3} = \frac{(u-t_0)N_{0,2}}{t_2-t_0} + \frac{(t_3-u)N_{1,2}}{t_3-t_1} = \frac{uN_{0,2}}{0} + \frac{(1-u)N_{1,2}}{1} = (1-u)^2$$

$$N_{1,3} = \frac{(u-t_1)N_{1,2}}{t_3-t_1} + \frac{(t_4-u)N_{2,2}}{t_4-t_2} = u(1-u) + (1-u)u = 2u(1-u)$$

$$N_{2,3} = \frac{(u-t_2)N_{2,2}}{t_4-t_2} + \frac{(t_5-u)N_{3,2}}{t_5-t_3} = u^2$$

다음 식을 위의 값을 이용하여 정리하면 다음과 같다.

$$P(u) = \sum_{i=0}^2 P_i N_{i,3}(u) \quad (t_2 =) 0 \leq u \leq 1 (= t_3)$$

$$P(u) = (1-u)^2 P_0 + 2u(1-u)P_1 + u^2 P_2$$

Rational Curves

- General Meaning
 - Functions are obtained by the “ratio” of two polynomials
 - This representation makes use of the concept of homogeneous coordinates
 - 한 꼭지점이 곡선에 미치는 영향의 양을 결정할 수 있음
- General Form

	Bezier	B-Spline
Nonrational (Integral)	$Q(t) = \sum_{i=0}^n B_{in}(t) V_i$	$P(t) = \sum_{i=0}^n N_{ik}(t) V_i$
Rational	$Q(t) = \frac{\sum_{i=0}^n B_{in}(t) w_i V_i}{\sum_{i=0}^n B_{in}(t) w_i}$	$P(t) = \frac{\sum_{i=0}^n N_{ik}(t) w_i V_i}{\sum_{i=0}^n N_{ik}(t) w_i}$

Rational Polynomial Curve

- Unit circle

- Implicit form: $x^2 + y^2 = 1$

- Parametric form

- $\mathbf{r}(u) = (\cos(u), \sin(u), 0)$
 - rational polynomial form

$$t = \tan(u / 2) \rightarrow \mathbf{r}(t) = \left(\frac{1-t^2}{1+t^2}, \frac{2t}{1+t^2}, 0 \right)$$

- polynomial form in homogeneous coordinates

$$\mathbf{R}(t) = (xw, yw, zw, w) = (1-t^2, 2t, 0, 1+t^2)$$

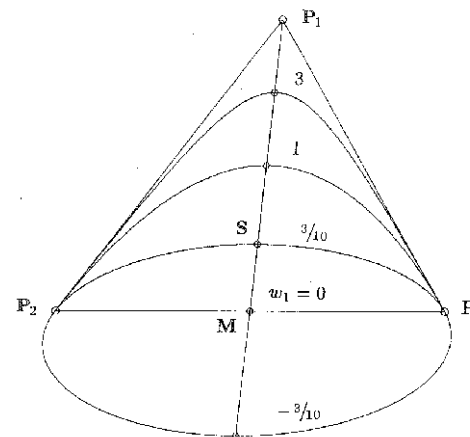
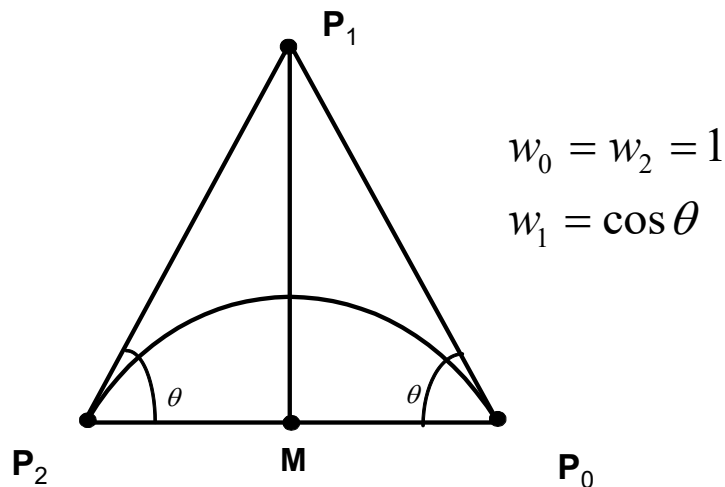
- Homogeneous coordinates에서 polynomial로 정확히 표현 가능

[Half Angle Formula]

$$\tan \frac{A}{2} = \pm \sqrt{\frac{1 - \cos A}{1 + \cos A}} = \frac{\sin A}{1 + \cos A}$$

Quadratic Rational Polynomial Curve

- n차 polynomial curve
 - 차수를 아무리 높여도 conic curve를 근사적으로 밖에 표현하지 못함
- Rational polynomial curve
 - 2차(quadratic)로 모든 종류의 conic curve를 정확히 표현함
 - NURB(Non-Uniform Rational B-spline)가 널리 쓰이는 이유



$$w_0 = w_2 = 1,$$

$$\begin{cases} w_1 = 1 & \text{parabola} \\ 0 \leq w_1 \leq 1 & \text{ellipse} \\ w_1 > 1 & \text{hyperbola} \end{cases}$$

NURBS Curve

- Non-Uniform Rational B-spline Curve
- 가장 일반적 형태의 B-spline curve
- NURB curve data (in IGES)

p : degree

n : highest index of control points ($= number - 1$)

$\mathbf{P}_0, \mathbf{P}_1, \dots, \mathbf{P}_n$: Euclidean control points

$w_0, w_1, \dots, w_n$: weights

$t_0, t_1, \dots, t_m$: knot vector ($m = n + p + 1$)

s_0, s_1 : start and end parameter values ($t_0 \leq s_0 < s_1 \leq t_m$)

- 참고사항 $\left\{ \begin{array}{l} \text{planar or nonplanar} \\ \text{open or closed} \\ \text{rational or nonrational} \\ \text{nonperiodic(clamped) or periodic(unclamped)} \end{array} \right.$

Quadratic Rational B-Spline

$k = 3$ (order)

$n = 2$ (# of control points $(= n + 1) = 3$)

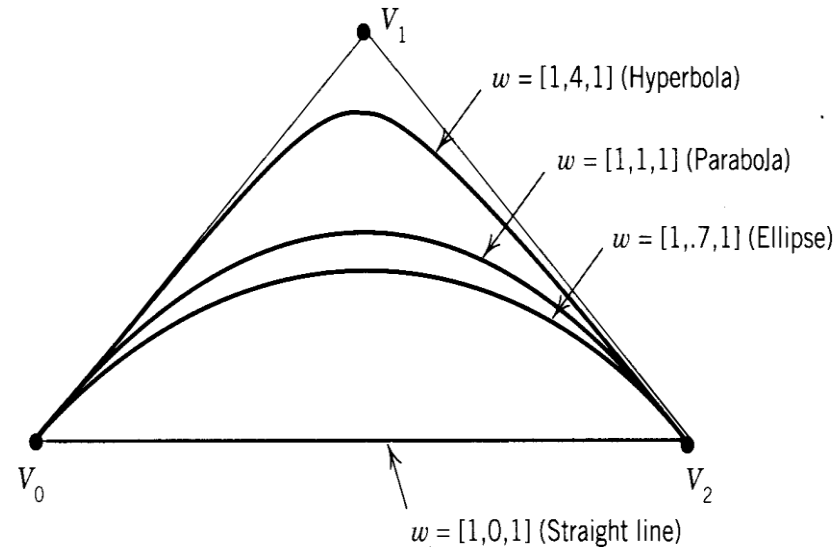
$m = n + k = 5$

knot vector : $\{0, 0, 0, 1, 1, 1\}$

weight : $\{1, w_1, 1\}$

$$\mathbf{P}(u) = \frac{w_0 \mathbf{V}_0 N_{0,3}(u) + w_1 \mathbf{V}_1 N_{1,3}(u) + w_2 \mathbf{V}_2 N_{2,3}(u)}{w_0 N_{0,3}(u) + w_1 N_{1,3}(u) + w_2 N_{2,3}(u)}$$

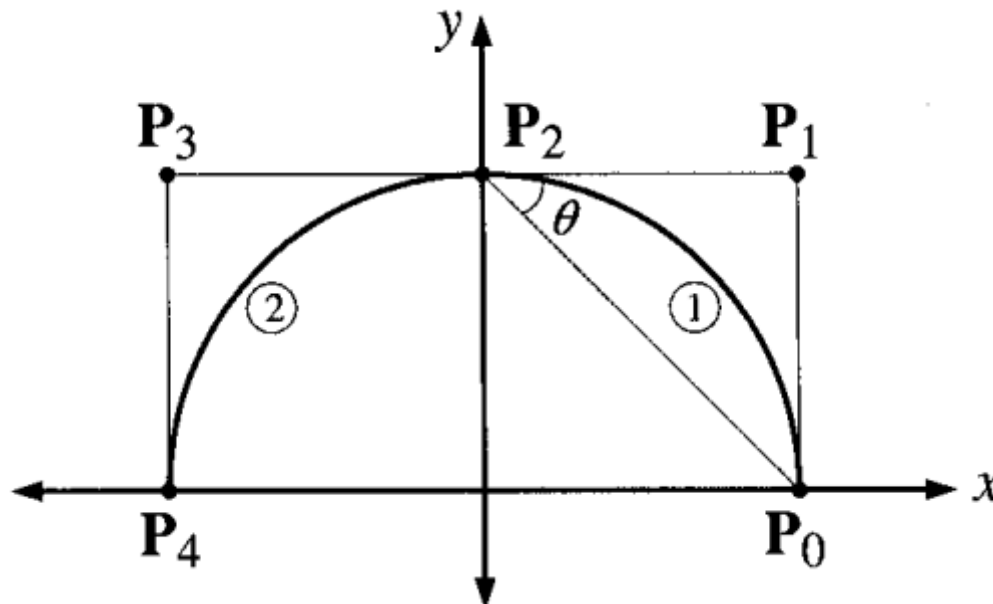
$$= \frac{\mathbf{V}_0 N_{0,3}(u) + w_1 \mathbf{V}_1 N_{1,3}(u) + \mathbf{V}_2 N_{2,3}(u)}{N_{0,3}(u) + w_1 N_{1,3}(u) + N_{2,3}(u)}$$



$w_1 = 0$	straight line
$0 \leq w_1 \leq 1$	ellipse segment
$w_1 = \cos \theta$	circle
$w_1 = 1$	parabola segment
$w_1 > 1$	hyperbola segment

Example

- Derive a NURB representation of a half circle of radius 1 in the xy plane.
- Expand the NURBS equation of arc 1 and show that it represents the circular arc exactly.



Example

degree = 2, order $k = 3$

half circle $\rightarrow$ two circular arcs (less than 180°)

$$\left. \begin{array}{l} \text{arc 1} \left\{ \begin{array}{l} \mathbf{P}_0 = (1,0), \mathbf{P}_1 = (1,1), \mathbf{P}_2 = (0,1) \\ w_0 = 1, w_1 = \cos 45^\circ = \frac{1}{\sqrt{2}}, w_2 = 1 \\ \text{knot} : 0,0,0,1,1,1 \end{array} \right. \\ \text{arc 2} \left\{ \begin{array}{l} \mathbf{P}_2 = (0,1), \mathbf{P}_3 = (-1,1), \mathbf{P}_4 = (-1,0) \\ w_2 = 1, w_3 = \cos 45^\circ = \frac{1}{\sqrt{2}}, w_4 = 1 \\ \text{knot} : 1,1,1,2,2,2 \end{array} \right. \end{array} \right\} \rightarrow \left\{ \begin{array}{l} \mathbf{P}_0 = (1,0), \mathbf{P}_1 = (1,1), \mathbf{P}_2 = (0,1), \mathbf{P}_3 = (-1,1), \mathbf{P}_4 = (-1,0) \\ w_0 = 1, w_1 = \frac{1}{\sqrt{2}}, w_2 = 1, w_3 = \frac{1}{\sqrt{2}}, w_4 = 1 \\ \text{knot} : 0,0,0,1,1,2,2,2 \end{array} \right.$$

$$\mathbf{P}(u) = \frac{\sum_{i=0}^n w_i \mathbf{P}_i N_{i,k}(u)}{\sum_{i=0}^n w_i N_{i,k}(u)} = \frac{w_0 \mathbf{P}_0 N_{0,3}(u) + w_1 \mathbf{P}_1 N_{1,3}(u) + w_2 \mathbf{P}_2 N_{2,3}(u)}{w_0 N_{0,3}(u) + w_1 N_{1,3}(u) + w_2 N_{2,3}(u)}$$

Example

$$N_{i,k}(u) = \frac{(u-t_i)N_{i,k-1}(u)}{t_{i+k-1}-t_i} + \frac{(t_{i+k}-u)N_{i+1,k-1}(u)}{t_{i+k}-t_{i+1}} \quad N_{i,1}(u) = \begin{cases} 1 & t_i \leq u \leq t_{i+1} \\ 0 & \text{otherwise} \end{cases}$$

$$t_0 = 0, t_1 = 0, t_2 = 0, t_3 = 1, t_4 = 1, t_5 = 1$$

$$N_{0,3}(u) = \frac{(u-t_0)N_{0,2}(u)}{t_2-t_0} + \frac{(t_3-u)N_{1,2}(u)}{t_3-t_1} = (1-u)^2$$

$$N_{1,2}(u) = \frac{(u-t_1)N_{1,1}(u)}{t_2-t_1} + \frac{(t_3-u)N_{2,1}(u)}{t_3-t_2} = 1-u$$

$$N_{2,1}(u) = \begin{cases} 1 & t_2 \leq u \leq t_3 \\ 0 & \text{otherwise} \end{cases}$$

$$N_{1,3}(u) = \frac{(u-t_1)N_{1,2}(u)}{t_3-t_1} + \frac{(t_4-u)N_{2,2}(u)}{t_4-t_2} = u(1-u) + (1-u)u = 2u(1-u)$$

$$N_{2,2}(u) = \frac{(u-t_2)N_{2,1}(u)}{t_3-t_2} + \frac{(t_4-u)N_{3,1}(u)}{t_4-t_3} = u$$

$$N_{2,3}(u) = \frac{(u-t_2)N_{2,2}(u)}{t_4-t_2} + \frac{(t_5-u)N_{3,2}(u)}{t_5-t_3} = u^2$$

Example

$$\mathbf{P}(u) = \frac{h_0 \mathbf{P}_0 N_{0,3}(u) + h_1 \mathbf{P}_1 N_{1,3}(u) + h_2 \mathbf{P}_2 N_{2,3}(u)}{h_0 N_{0,3}(u) + h_1 N_{1,3}(u) + h_2 N_{2,3}(u)} = \frac{1 \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix} (1-u)^2 + \frac{1}{\sqrt{2}} \begin{pmatrix} 1 \\ 1 \\ 0 \end{pmatrix} 2u(1-u) + 1 \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix} u^2}{1(1-u)^2 + \frac{1}{\sqrt{2}} 2u(1-u) + 1u^2}$$

$$\left. \begin{aligned} x(u) &= \frac{(1-\sqrt{2})u^2 + \sqrt{2}(1-\sqrt{2})u + 1}{(2-\sqrt{2})u^2 + (\sqrt{2}-2)u + 1} \\ y(u) &= \frac{(1-\sqrt{2})u^2 + \sqrt{2}u}{(2-\sqrt{2})u^2 + (\sqrt{2}-2)u + 1} \end{aligned} \right\}$$

$$\rightarrow [x(u)]^2 + [y(u)]^2 = \frac{\left[(1-\sqrt{2})u^2 + \sqrt{2}(1-\sqrt{2})u + 1 \right]^2 + \left[(1-\sqrt{2})u^2 + \sqrt{2}u \right]^2}{\left[(2-\sqrt{2})u^2 + (\sqrt{2}-2)u + 1 \right]^2} = 1$$

How to Choose a Spline

- Hermite curves are good for single segments where you know the parametric derivative or want easy control of it
- Bezier curves are good for single segments or patches where a user controls the points
- B-splines are good for large continuous curves and surfaces
- NURBS are the most general, and are good when that generality is useful, or when conic sections must be accurately represented (CAD)

Surfaces

- Bilinear Surface $\leftarrow$ four points
- Coon's Patch $\leftarrow$ four boundary curves
- Bicubic Patch $\leftarrow$ geometric form
- Bezier Surface
- B-Spline Surface
- NURBS Surface

Bilinear Surface

- A bilinear surface is derived by interpolating the four data points with the linear equations in the parameters u and v such that the resulting surface has the four points at its corners

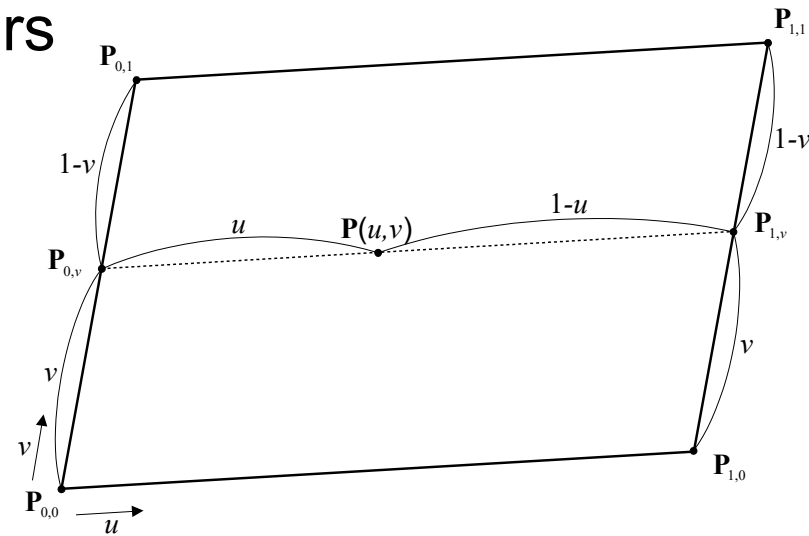
$$\mathbf{P}_{0,v} = (1-v)\mathbf{P}_{0,0} + v\mathbf{P}_{0,1}$$

$$\mathbf{P}_{1,v} = (1-v)\mathbf{P}_{1,0} + v\mathbf{P}_{1,1}$$

$$\mathbf{P}(u,v) = (1-u)\mathbf{P}_{0,v} + u\mathbf{P}_{1,v}$$

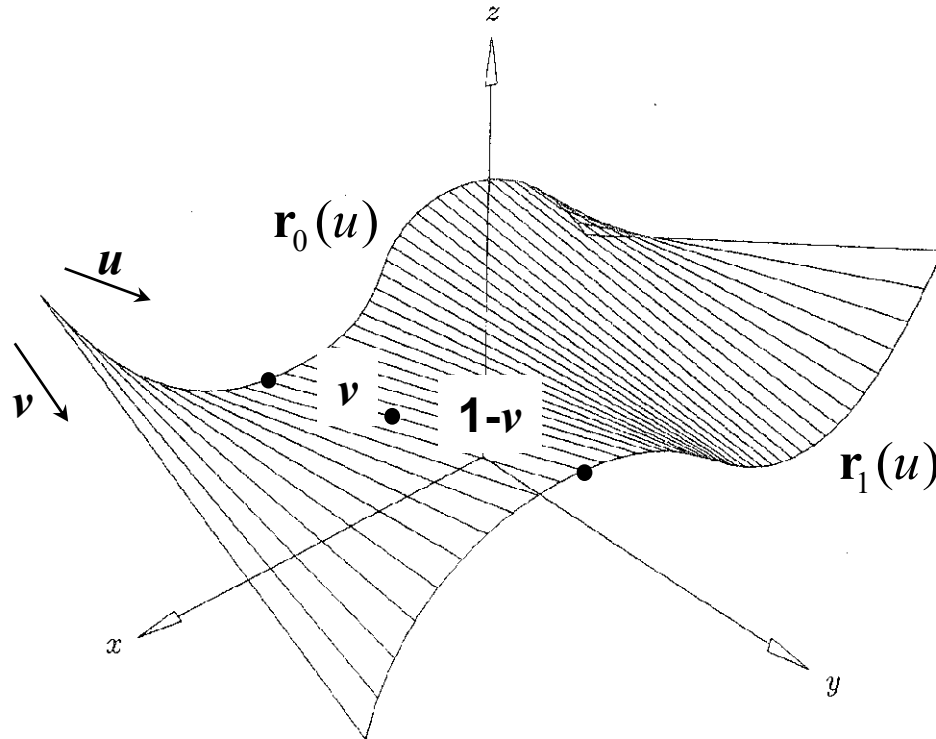
$$\mathbf{P}(u,v) = (1-u)\left[(1-v)\mathbf{P}_{0,0} + v\mathbf{P}_{0,1}\right] + u\left[(1-v)\mathbf{P}_{1,0} + v\mathbf{P}_{1,1}\right]$$

$$= \begin{bmatrix} (1-u)(1-v) & u(1-v) & (1-u)v & uv \end{bmatrix} \begin{bmatrix} \mathbf{P}_{0,0} \\ \mathbf{P}_{1,0} \\ \mathbf{P}_{0,1} \\ \mathbf{P}_{1,1} \end{bmatrix} \quad \begin{pmatrix} 0 \leq u \leq 1 \\ 0 \leq v \leq 1 \end{pmatrix}$$

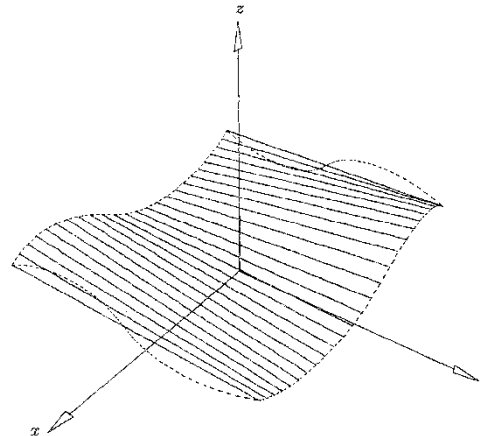
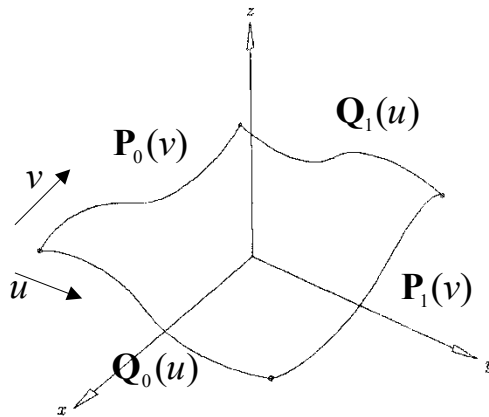


Ruled Surface

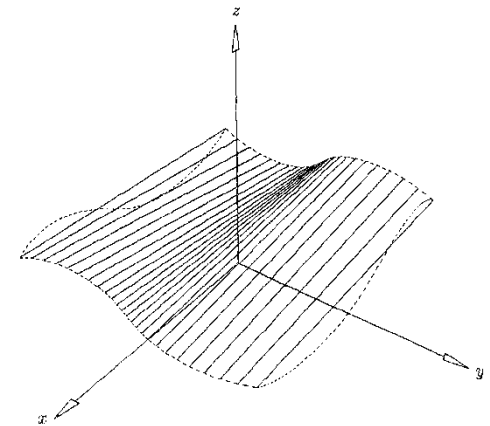
$$\mathbf{r}(u, v) = (1 - v)\mathbf{r}_0(u) + v\mathbf{r}_1(u), \quad 0 \leq v \leq 1$$



Coon's Patch: 4 boundary curves

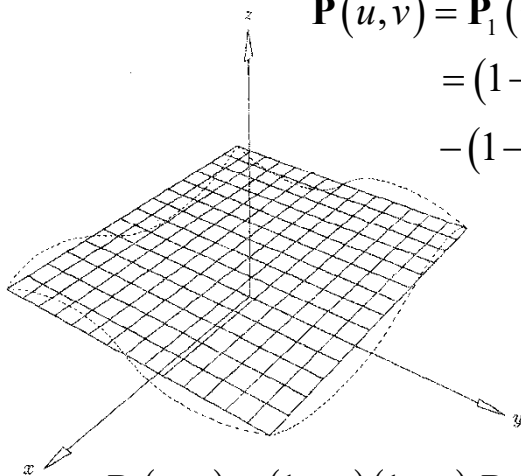


$$\mathbf{P}_1(u, v) = (1-u)\mathbf{P}_0(v) + u\mathbf{P}_1(v)$$

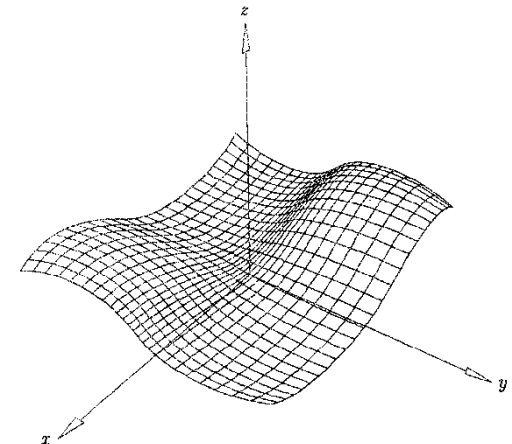


$$\mathbf{P}_2(u, v) = (1-v)\mathbf{Q}_0(u) + v\mathbf{Q}_1(u)$$

$$\begin{aligned} \mathbf{P}(u, v) &= \mathbf{P}_1(u, v) + \mathbf{P}_2(u, v) - \mathbf{P}_3(u, v) \\ &= (1-u)\mathbf{P}_0(v) + u\mathbf{P}_1(v) + (1-v)\mathbf{Q}_0(u) + v\mathbf{Q}_1(u) \\ &\quad - (1-u)(1-v)\mathbf{P}_{0,0} - u(1-v)\mathbf{P}_{1,0} - (1-u)v\mathbf{P}_{0,1} - uv\mathbf{P}_{1,1} \\ &\quad 0 \leq u \leq 1, 0 \leq v \leq 1 \end{aligned}$$



$$\mathbf{P}_3(u, v) = (1-u)(1-v)\mathbf{P}_{0,0} + u(1-v)\mathbf{P}_{1,0} + (1-u)v\mathbf{P}_{0,1} + uv\mathbf{P}_{1,1}$$



Bicubic Patch (1)

- Bicubic Patch
 - Extension of the parametric cubic curve formulation
 - Boundary curves are parametric cubics or Hermites
 - The Interior is defined by blending functions
- Algebraic form

$$\mathbf{P}(u, v) = \sum_{i=0}^3 \sum_{j=0}^3 \mathbf{a}_{ij} u^i v^j \quad 0 \leq u \leq 1, 0 \leq v \leq 1$$

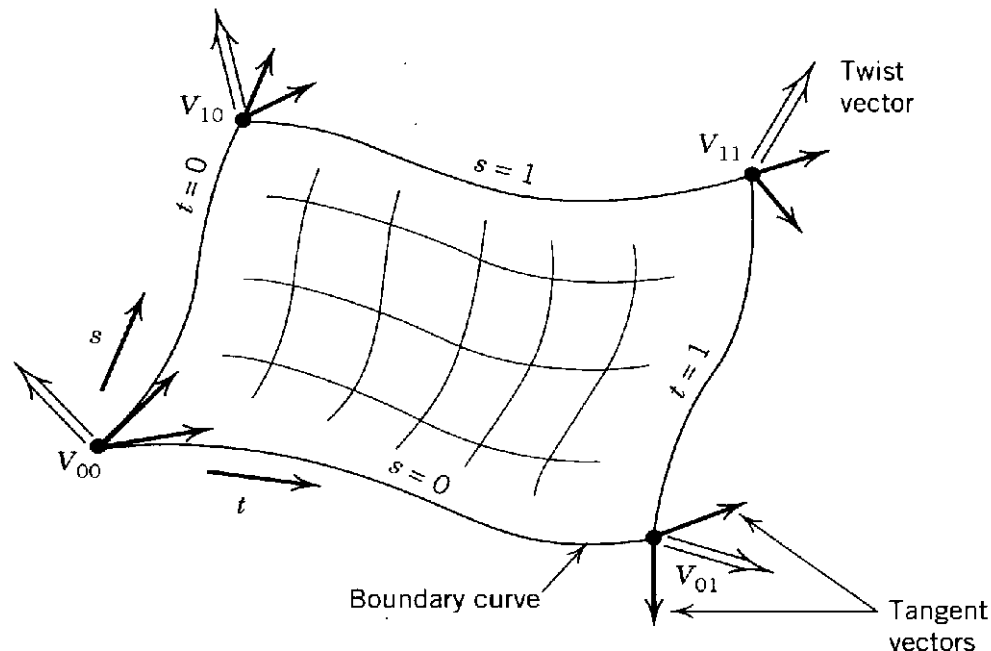
- Matric form

$$\mathbf{P}(u, v) = \begin{bmatrix} 1 & u & u^2 & u^3 \end{bmatrix} \begin{bmatrix} \mathbf{a}_{00} & \mathbf{a}_{01} & \mathbf{a}_{02} & \mathbf{a}_{03} \\ \mathbf{a}_{10} & \mathbf{a}_{11} & \mathbf{a}_{12} & \mathbf{a}_{13} \\ \mathbf{a}_{20} & \mathbf{a}_{21} & \mathbf{a}_{22} & \mathbf{a}_{23} \\ \mathbf{a}_{30} & \mathbf{a}_{31} & \mathbf{a}_{32} & \mathbf{a}_{33} \end{bmatrix} \begin{bmatrix} 1 \\ v \\ v^2 \\ v^3 \end{bmatrix}$$

Bicubic Patch (2)

- Geometric Form

- 16 vector equations are required for 16 algebraic coefficient vectors
- 16 Boundary conditions
 - 4 corner points: $P(0,0)$, $P(0,1)$, $P(1,0)$, $P(1,1)$
 - 8 tangent vectors at corner points: $P_u(0,0)$, $P_u(0,1)$, $P_u(1,0)$, $P_u(1,1)$, $P_v(0,0)$, $P_v(0,1)$, $P_v(1,0)$, $P_v(1,1)$
 - 4 twist vectors at corner points: $P_{uv}(0,0)$, $P_{uv}(0,1)$, $P_{uv}(1,0)$, $P_{uv}(1,1)$



Bicubic Patch (3)

$$\mathbf{P}(u, v) = \begin{bmatrix} F_1(u) & F_2(u) & F_3(u) & F_4(u) \end{bmatrix} \begin{bmatrix} \mathbf{P}(0,0) & \mathbf{P}(0,1) & \mathbf{P}_v(0,0) & \mathbf{P}_v(0,1) \\ \mathbf{P}(1,0) & \mathbf{P}(1,1) & \mathbf{P}_v(1,0) & \mathbf{P}_v(1,1) \\ \mathbf{P}_u(0,0) & \mathbf{P}_u(0,1) & \mathbf{P}_{uv}(0,0) & \mathbf{P}_{uv}(0,1) \\ \mathbf{P}_u(1,0) & \mathbf{P}_u(1,1) & \mathbf{P}_{uv}(1,0) & \mathbf{P}_{uv}(1,1) \end{bmatrix} \begin{bmatrix} F_1(v) \\ F_2(v) \\ F_3(v) \\ F_4(v) \end{bmatrix}$$

$$0 \leq u \leq 1, 0 \leq v \leq 1$$

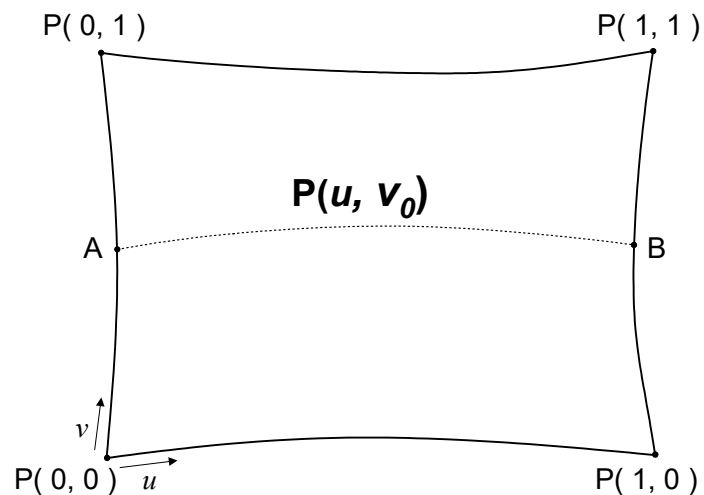
- Blending Functions (Hermite Curve Eqns)

$$F_1(u) = 1 - 3u^2 + 2u^3$$

$$F_2(u) = 3u^2 - 2u^3$$

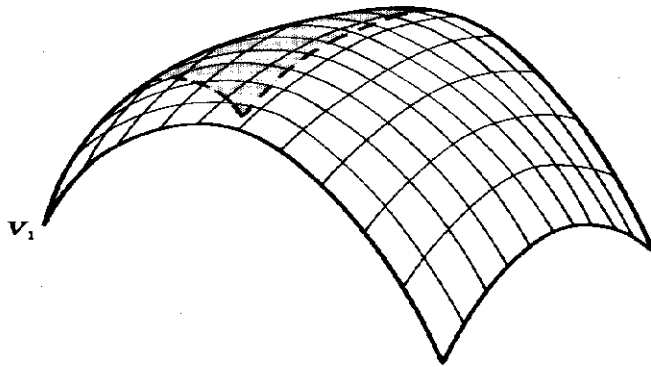
$$F_3(u) = u - 2u^2 + u^3$$

$$F_4(u) = -u^2 + u^3$$

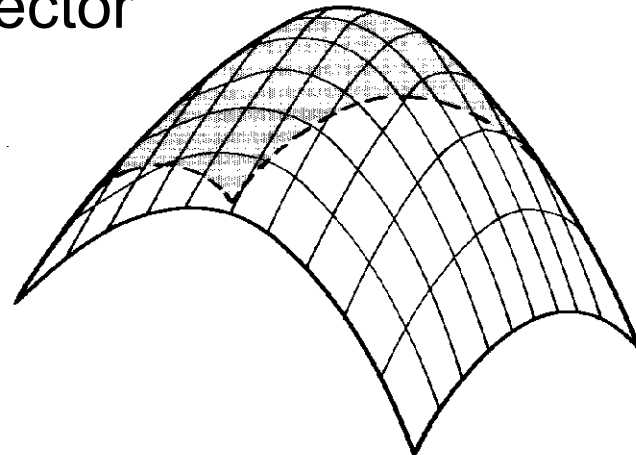


Bicubic Patch (4)

- Effect of Variation in Twist Vector

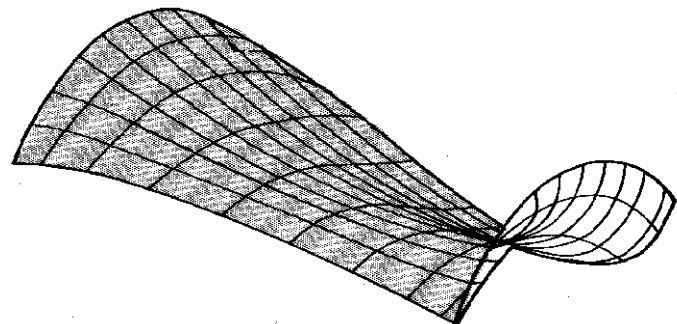
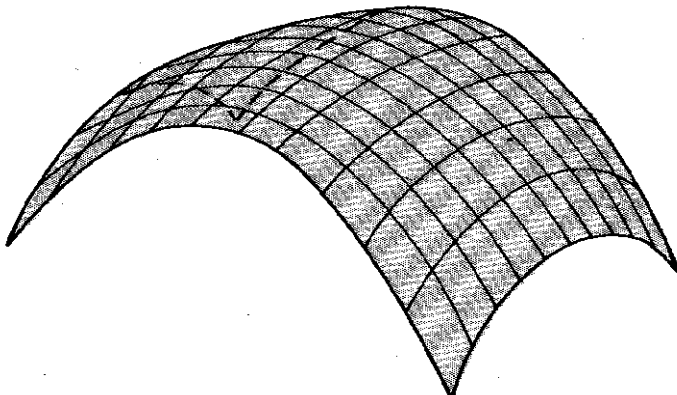


(a)



(b)

- Effect of Variation in Tangent Vector



Ferguson Patch (F-Patch)

- Definition
 - Setting all twist vector to zero
 - $P_{uv}(0,0) = P_{uv}(0,1) = P_{uv}(1,0) = P_{uv}(1,1) = 0$
 - Not commonly used in practice because they force the surface to flatten at the corners
- Disadvantage
 - No intuitive feel for the values of the tangent and twist vectors is available to the user

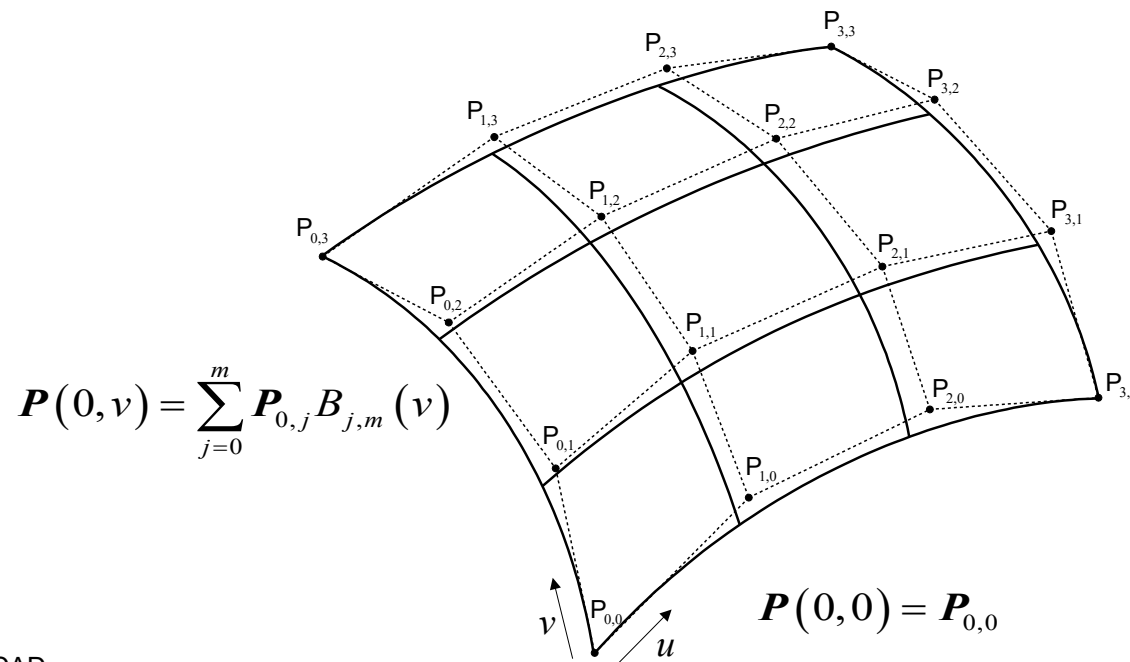
Bezier Surface

$$\begin{aligned} \mathbf{P}(u, v) &= \sum_{i=0}^n \sum_{j=0}^m \mathbf{P}_{i,j} B_{i,n}(u) B_{j,m}(v) \quad 0 \leq u \leq 1, \quad 0 \leq v \leq 1 \\ &= \sum_{i=0}^n \left[\mathbf{P}_{i,0} B_{0,m}(v) + \mathbf{P}_{i,1} B_{1,m}(v) + \cdots + \mathbf{P}_{i,m} B_{m,m}(v) \right] B_{i,n}(u) \end{aligned}$$

$\mathbf{P}_{i,j}$: control points

$B_{i,n}(u), B_{j,m}(v)$: Bernstein blending functions in the u and v directions

Note : n does not have to be the same as m



$\mathbf{P}_{1,0}, \mathbf{P}_{2,0}, \mathbf{P}_{1,3}, \mathbf{P}_{2,3}$

$\leftrightarrow \mathbf{P}_u(0,0), \mathbf{P}_u(1,0), \mathbf{P}_u(0,1), \mathbf{P}_u(1,1)$

$\mathbf{P}_{0,1}, \mathbf{P}_{0,2}, \mathbf{P}_{3,1}, \mathbf{P}_{3,2}$

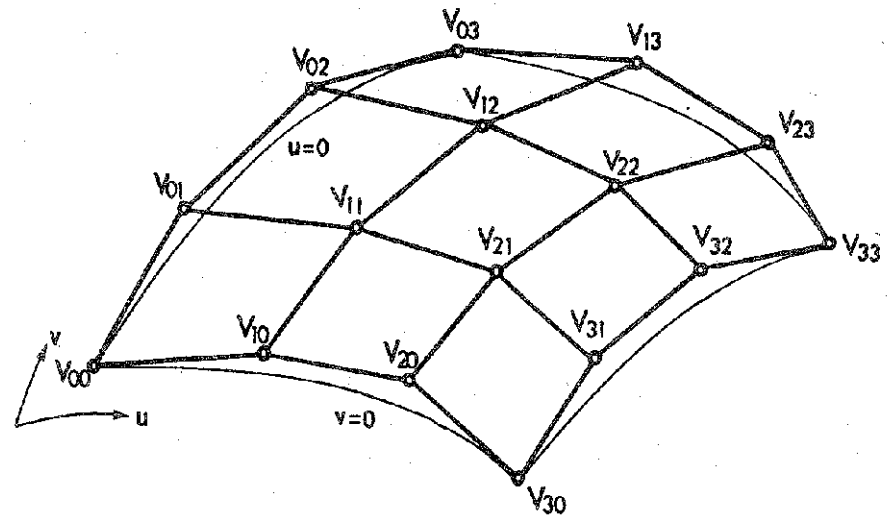
$\leftrightarrow \mathbf{P}_v(0,0), \mathbf{P}_v(0,1), \mathbf{P}_v(1,0), \mathbf{P}_v(1,1)$

$\mathbf{P}_{1,1}, \mathbf{P}_{2,1}, \mathbf{P}_{1,2}, \mathbf{P}_{2,2}$

$\leftrightarrow$ twisting vectors (internal shape)

Bicubic Bezier Surface

$$\begin{aligned}
 \mathbf{P}(u, v) &= \sum_{i=0}^3 \sum_{j=0}^3 B_i^3(u) B_j^3(v) \mathbf{V}_{ij} \\
 &= \sum_{i=0}^3 B_i^3(u) \left(\sum_{j=0}^3 B_j^3(v) \mathbf{V}_{ij} \right) \\
 &= \sum_{i=0}^3 B_i^3(u) \mathbf{b}_i(v) \\
 &= \mathbf{U} \mathbf{M} \mathbf{B}^T \mathbf{V}^T
 \end{aligned}$$

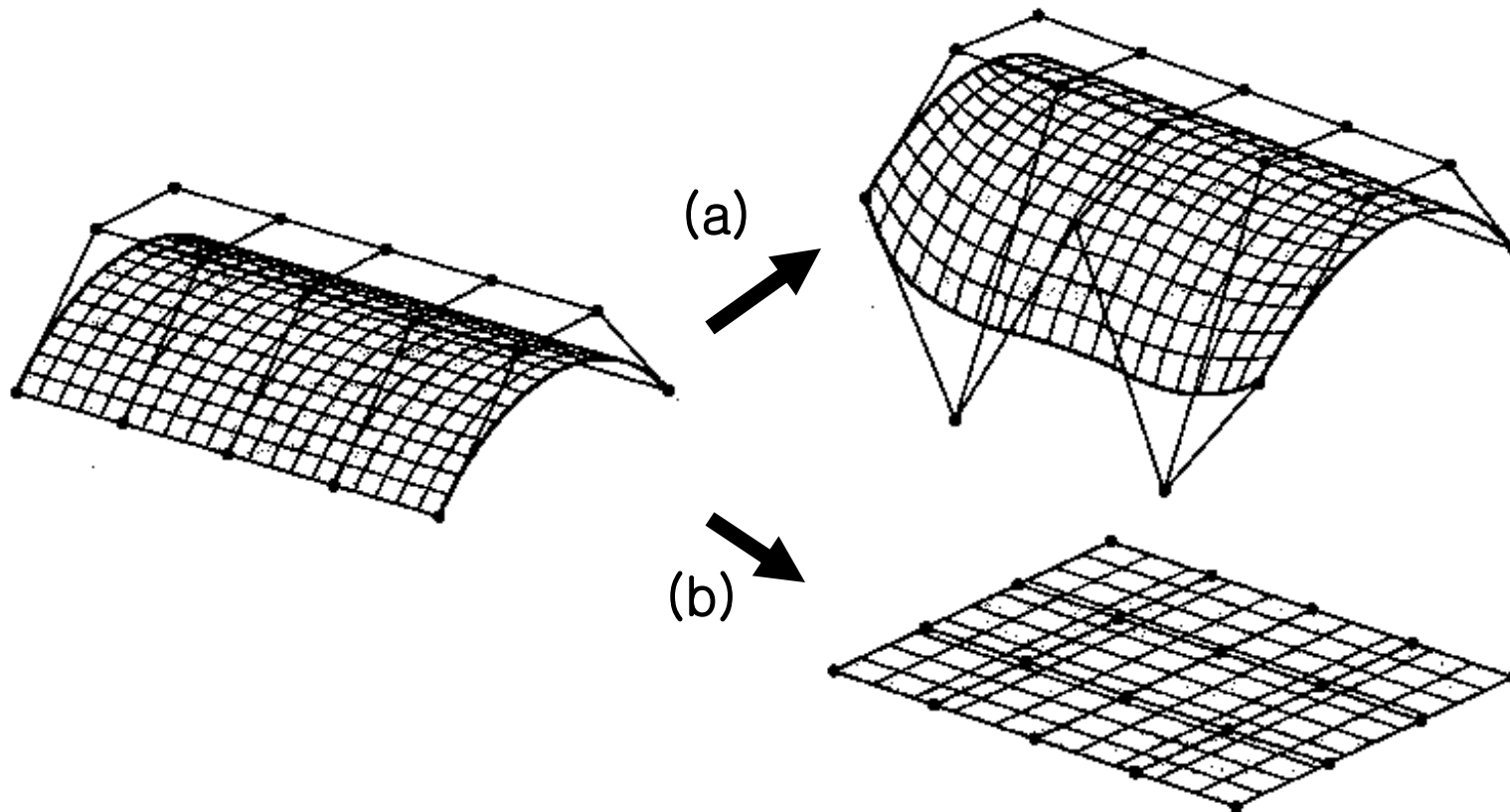


$$\mathbf{U} = \begin{bmatrix} 1 & u & u^2 & u^3 \end{bmatrix}, \mathbf{V} = \begin{bmatrix} 1 & v & v^2 & v^3 \end{bmatrix}$$

$$\mathbf{M} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ -3 & 3 & 0 & 0 \\ 3 & -6 & 3 & 0 \\ -1 & 3 & -3 & 1 \end{bmatrix}, \mathbf{B} = \begin{bmatrix} \mathbf{V}_{00} & \mathbf{V}_{01} & \mathbf{V}_{02} & \mathbf{V}_{03} \\ \mathbf{V}_{10} & \mathbf{V}_{11} & \mathbf{V}_{12} & \mathbf{V}_{13} \\ \mathbf{V}_{20} & \mathbf{V}_{21} & \mathbf{V}_{22} & \mathbf{V}_{23} \\ \mathbf{V}_{30} & \mathbf{V}_{31} & \mathbf{V}_{32} & \mathbf{V}_{33} \end{bmatrix}$$

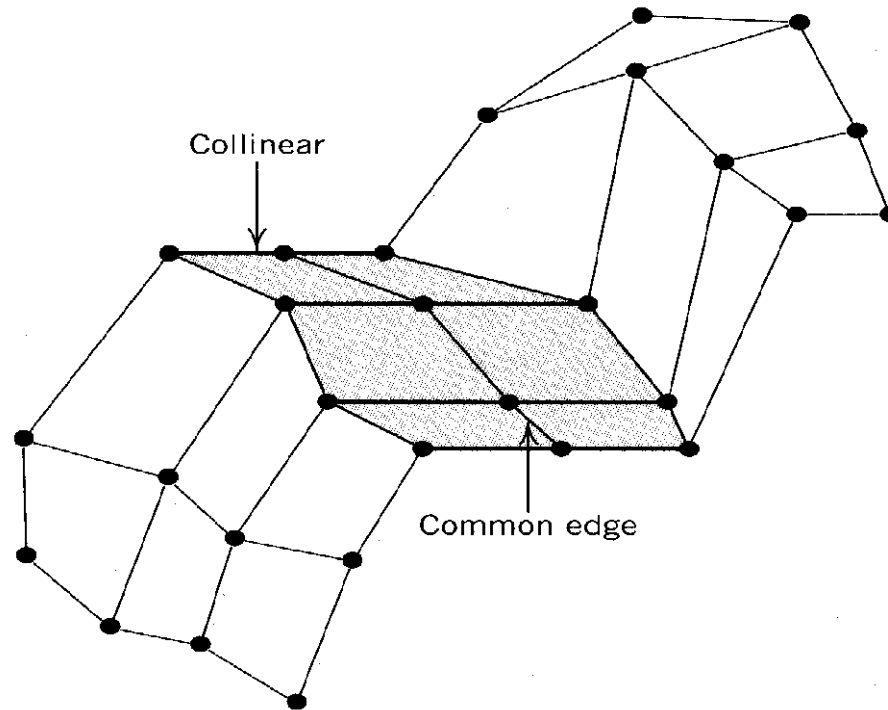
Effect of Moving Control Points

- (a) At the boundary curves
- (b) On the interior part of the surface



Continuity of Bicubic Bezier Surfaces

- First degree parametric continuity is enforced along the common edge between two patches



B-Spline Surface

$$\mathbf{P}(u, v) = \sum_{i=0}^n \sum_{j=0}^m \mathbf{P}_{i,j} N_{i,k}(u) N_{j,l}(v) \quad (s_{k-1} \leq u \leq s_{n+1}, t_{l-1} \leq v \leq t_{m+1})$$

$\mathbf{P}_{i,j}$: control points

$N_{i,k}(u), N_{j,l}(v)$: B-spline blending functions in the u and v directions

Note: k does not have to be the same as l

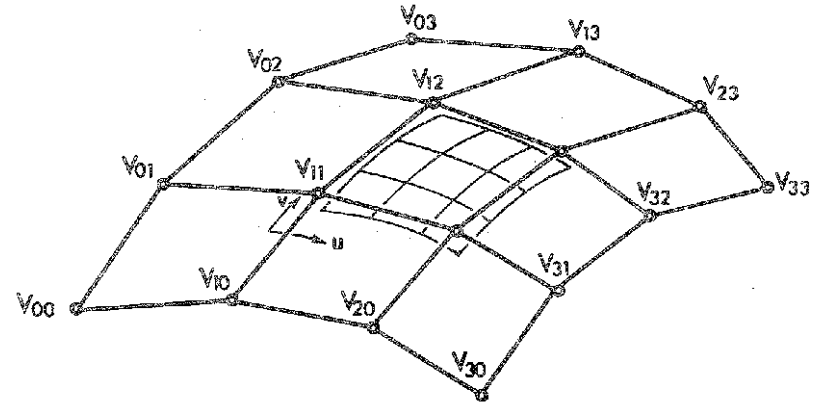
$$\mathbf{P}(0, v) = \sum_{j=0}^m \left[\sum_{i=0}^n \mathbf{P}_{i,j} N_{i,k}(u) \right]_{u=0} N_{j,l}(v) = \sum_{j=0}^m \mathbf{P}_{0,j} N_{j,l}(v) \leftarrow \left[\sum_{i=0}^n \mathbf{P}_i N_{i,k}(u) \right]_{u=0} = \mathbf{P}_0$$

$$\begin{aligned} \mathbf{P}(u_0, v_0) &= \sum_{i=0}^n \sum_{j=0}^m \mathbf{P}_{i,j} N_{i,k}(u_0) N_{j,l}(v_0) \\ &= \left[\sum_{j=0}^m \mathbf{P}_{0,j} N_{j,l}(v_0) \right] N_{0,k}(u_0) + \cdots + \left[\sum_{j=0}^m \mathbf{P}_{n,j} N_{j,l}(v_0) \right] N_{n,k}(u_0) \\ &= \mathbf{C}_0 N_{0,k}(u_0) + \mathbf{C}_1 N_{1,k}(u_0) + \cdots + \mathbf{C}_n N_{n,k}(u_0) \end{aligned}$$

Bicubic B-spline Surface (Uniform)

$$\mathbf{P}(u, v) = \sum_{i=0}^3 \sum_{j=0}^3 N_{i,3}(u) N_{j,3}(v) \mathbf{V}_{ij}$$

$$= \mathbf{U} \mathbf{B} \mathbf{N}^T \mathbf{V}^T \quad (0 \leq u, v \leq 1)$$

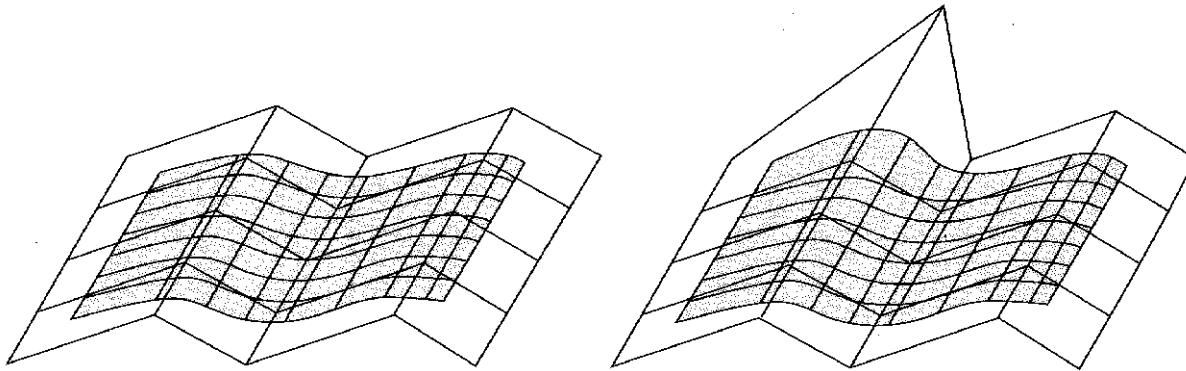


$$\mathbf{U} = \begin{bmatrix} 1 & u & u^2 & u^3 \end{bmatrix}, \quad \mathbf{V} = \begin{bmatrix} 1 & v & v^2 & v^3 \end{bmatrix}$$

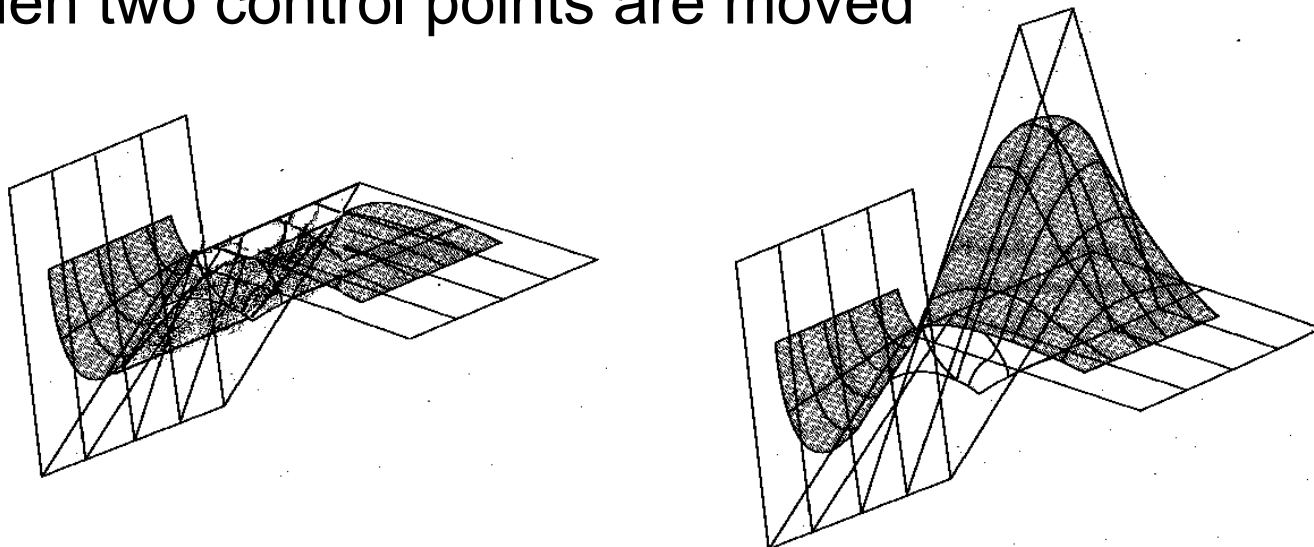
$$\mathbf{B} = \begin{bmatrix} \mathbf{V}_{00} & \mathbf{V}_{01} & \mathbf{V}_{02} & \mathbf{V}_{03} \\ \mathbf{V}_{10} & \mathbf{V}_{11} & \mathbf{V}_{12} & \mathbf{V}_{13} \\ \mathbf{V}_{20} & \mathbf{V}_{21} & \mathbf{V}_{22} & \mathbf{V}_{23} \\ \mathbf{V}_{30} & \mathbf{V}_{31} & \mathbf{V}_{32} & \mathbf{V}_{33} \end{bmatrix}, \quad \mathbf{N} = \frac{1}{6} \begin{bmatrix} 1 & 4 & 1 & 0 \\ -3 & 0 & 3 & 0 \\ 3 & -6 & 3 & 0 \\ -1 & 3 & -1 & 1 \end{bmatrix}$$

B-Spline Surface

- When one control point is moved
 - only a small portion of the B-spline surface is affected



- When two control points are moved^(α)^(h)



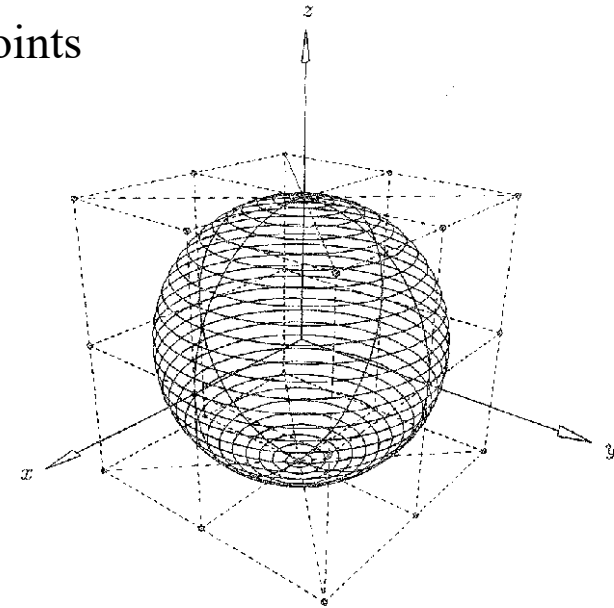
NURBS Surface (1)

$$\mathbf{P}(u, v) = \frac{\sum_{i=0}^n \sum_{j=0}^m h_{i,j} \mathbf{P}_{i,j} N_{i,k}(u) N_{j,l}(v)}{\sum_{i=0}^n \sum_{j=0}^m h_{i,j} N_{i,k}(u) N_{j,l}(v)} \quad \left(\begin{array}{c} s_{k-1} \leq u \leq s_{n+1} \\ t_{l-1} \leq v \leq t_{m+1} \end{array} \right)$$

$\mathbf{P}_{i,j}$: x , y and z coordinates of the control points

$h_{i,j}$: homogeneous coordinates of the control points

- Quadric (Quadratic) NURBS Surface로 Cylinder, Cone, Sphere, Paraboloid, Hyperboloid를 정확히 나타낼 수 있다.



NURBS Surface (2)

- Effect of Weights

- The weights provide an additional degree of freedom for the shape of surface
- Larger values of weights at the interior control points
- Lower values of weights at the top interior control points

