

# コンピュータグラフィックス 基礎

## 第6回 曲線・曲面の表現 「Bスプライン曲線」

金森 由博

# 学習の目標

- B スプライン曲線を学ぶ
- 曲線の基底関数の概念を理解する
- 制御点を入力することで、B スプライン曲線を描画するアプリケーションの開発を行えるようになる

# 基底関数の理解

曲線上の点の位置は、パラメータ  $t$  と制御点の位置によって定義される

曲線上の点の位置は、制御点の座標を混ぜ合わせて作る

$$P(t) = a(t)P_0 + \beta(t)P_1 + \gamma(t)P_2 \cdot \cdot \cdot$$

この混ぜ合わせ方の係数関数  $\alpha$ 、 $\beta$ 、 $\gamma$   $\cdot \cdot \cdot$  を定義したものが基底関数

# ベジエ曲線の復習

$n$  次ベジエ曲線（ベジエ曲線の一般化）

$$P(t) = \sum_{i=0}^n \boxed{{}_n C_i t^i (1-t)^{n-i}} P_i$$

↑  
制御点に対する係数  
(重み付け、混合比)

パラメータ  $t$  によって変化する

$${}_n C_i = \frac{n!}{i!(n-i)!}$$

↑  
2項係数

**「基底関数」**

# ベジエ曲線の復習

## • バーンスタイン基底関数

$$B_i^n = \binom{n}{i} C_i t^i (1-t)^{n-i}$$

$n$  は次数を  $i$  は制御点の番号を表す

■ 図3.24——3次バーンスタイン基底関数のグラフ

3次の場合

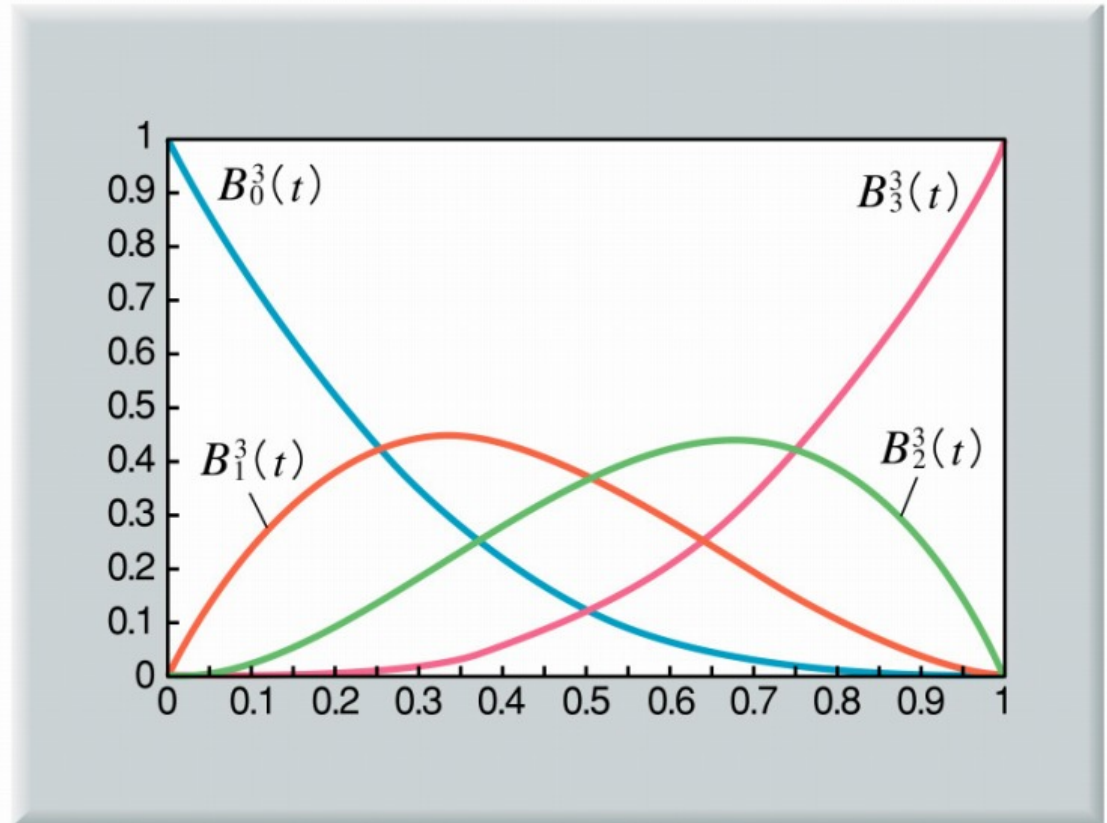
$$B_0^3 = (1-t)^3$$

$$B_1^3 = 3t(1-t)^2$$

$$B_2^3 = 3t^2(1-t)$$

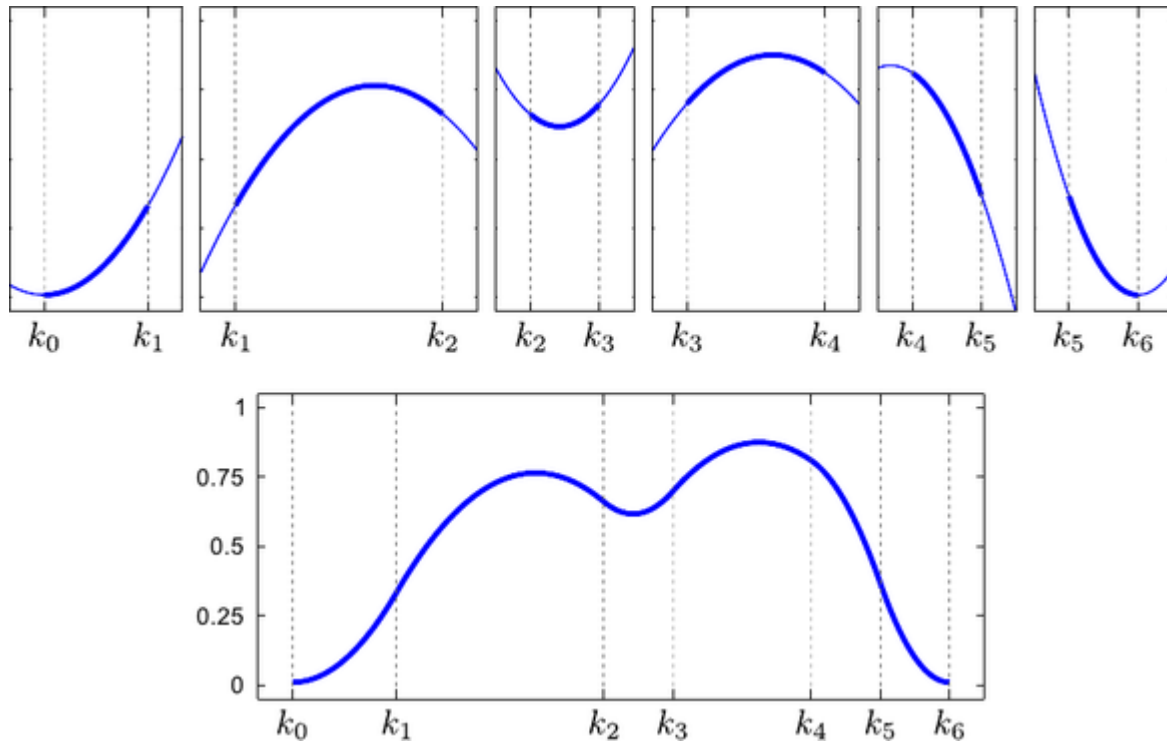
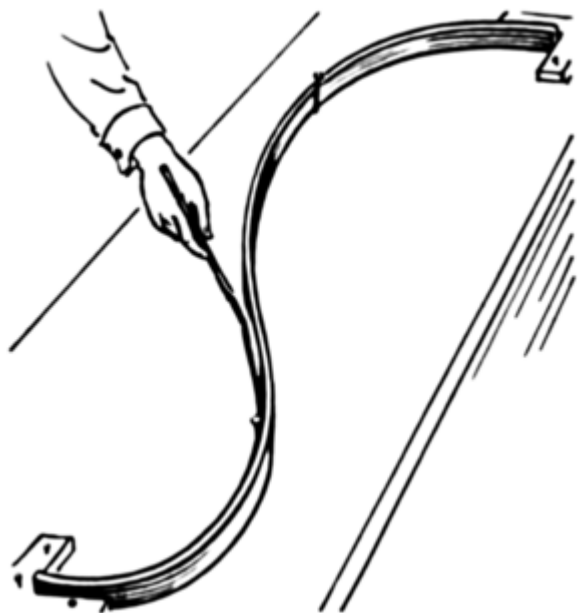
$$B_3^3 = t^3$$

3次のベジエ曲線において  
制御点  $P_0$  の重み付けが  
どのように変化するか



# B スプライン曲線

# スプライン曲線のイメージ



複数の曲線セグメントを繋ぎ合わせて作る  
物理的には最も曲げエネルギーの小さい曲線

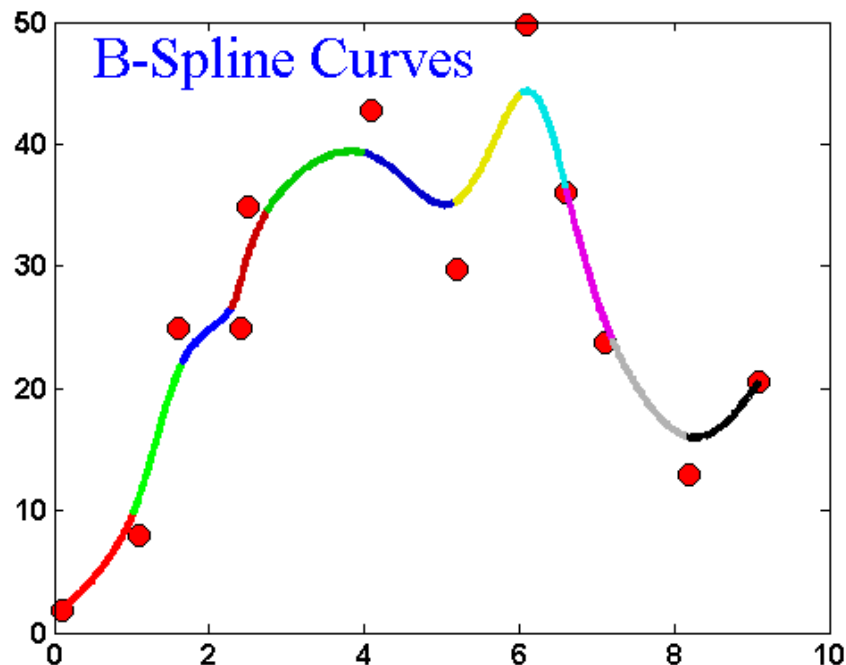
# B スプライン曲線の特徴

- セグメントが常に微分も含めて連続的に接続する  
(ベジエ曲線ではセグメント間の微分の連続性は保証されていない)
- 制御点をいくつでも指定できる  
※ 曲線上の1点に着目した場合、その点の位置に影響を与えるのはベジエ曲線と同じ[次数 + 1]個の制御点
- パラメータ  $t$  の値は 0 から 1 の範囲に限らない
- ベジエ曲線を表現できる



# B スプライン曲線の形を決めるもの

- 制御点列  $(P_i)$  + ノット列 (ノットベクトル)  $(t_j)$
- 複数の多項式曲線 (セグメント) を接続して 1 本の曲線とする

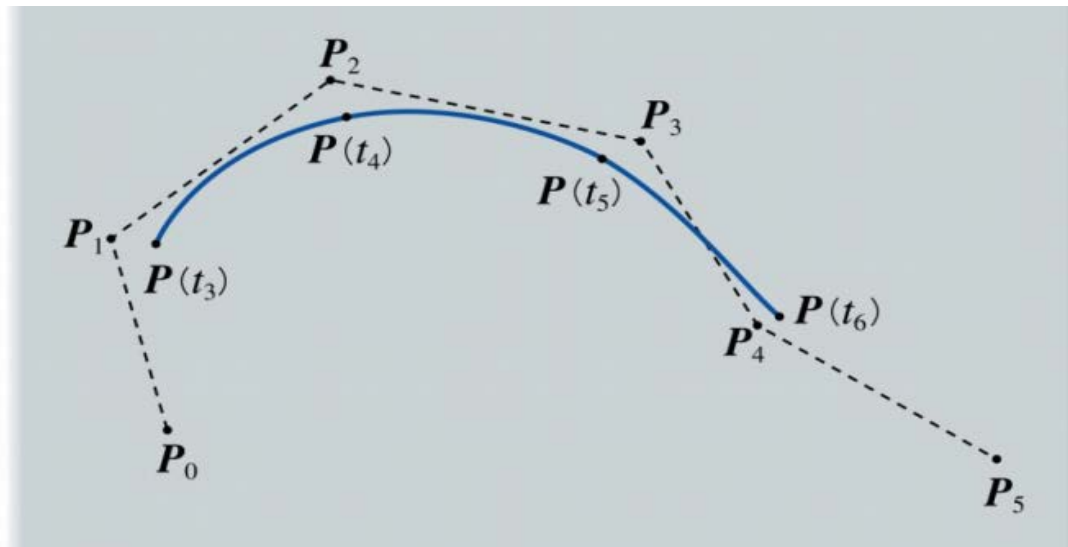


# B スプラインの形を決めるもの

- 「ノット列」は接続点でのパラメータ  $t$  の値の列。ノット列の値は単調増加  $t_i \leq t_{i+1}$

パラメータ  $t$  の範囲は  $t_n$  から  $t_{n+L}$  まで。(  $n$  は次数、  $L$  はセグメント数 )

- (制御点の数) = (次数) + (セグメント数)
- (ノット数) =  $2 \times$  (次数) + (セグメント数 + 1)
- 下の例は 次数 3、セグメント数 3、制御点数 6、ノット数 10  
( $t_0, t_1, t_2, t_3, t_4, t_5, t_6, t_7, t_8, t_9$ ) = (-3, -2, -1, 0, 1, 2, 3, 4, 6, 7)



# 練習問題

- 3 次の B スプライン曲線について
  1. セグメント数が 1 のとき、制御点の数とノット列の要素数はいくつか
  2. セグメント数が 3 のときは、それぞれいくつか
  3. 制御点の数が 7 のとき、ノット列とセグメント数はそれぞれいくつか
  4. 制御点の数が 10 のときは、それぞれいくつか

# B スプラインの数式表現

$$P(t) = \sum_{i=0}^{n+L-1} \underline{N_i^n(t)} P_i$$

混合関数 「Bスプライン基底関数」

$n$ : 次数

$L$ : セグメントの数

制御点の数は  $n+L$  (3 次でセグメント数 1 なら  
制御点は 4 つ)

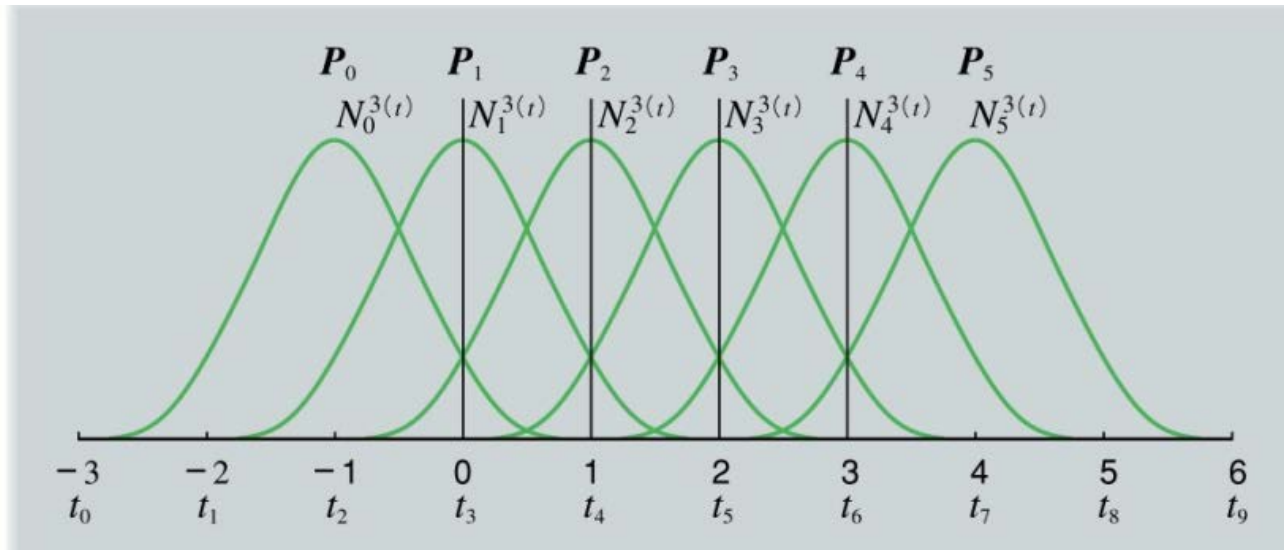
$$N_i^0(t) = \begin{cases} 1 & (t_i \leq t < t_{i+1}) \\ 0 & \text{上記以外} \end{cases}$$

$$N_i^n(t) = \frac{t - t_i}{t_{i+n} - t_i} N_i^{n-1}(t) + \frac{t_{i+n+1} - t}{t_{i+n+1} - t_{i+1}} N_{i+1}^{n-1}(t)$$

基底関数は**再帰的**に求まる

手計算は大変だけどプログラムなら再帰関数ですぐ求まる

# 基底関数のグラフ（一様3次）

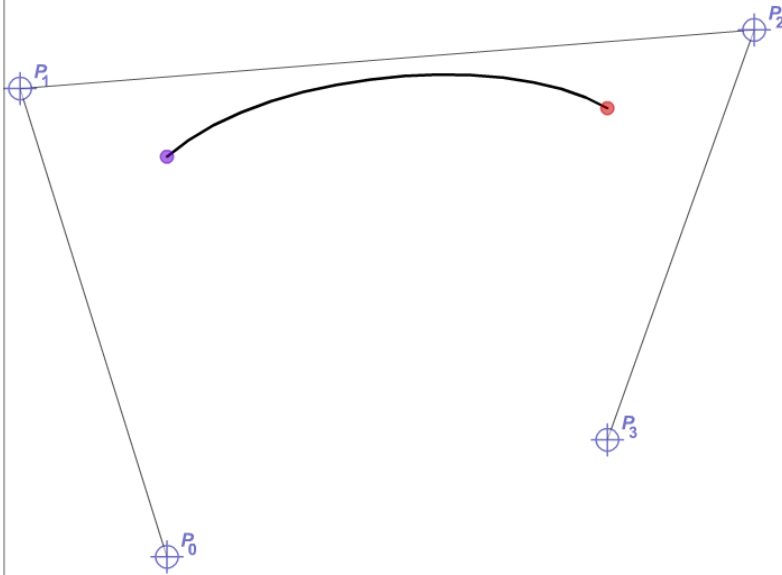


- ノットベクトル  $t_i$  が一定の間隔で存在 → 一様  
(ノットの間隔を変更すると「非一様」になる)
- ある  $t$  の値では 4 つのグラフが存在 → 4 つの制御点が影響
- $P(t_4)$  から  $P(t_5)$  は、 $P_0, P_5$  の影響を受けない → 局所性
- 最初と最後のノットを「次数+1」個だけ重ねると、端点が制御点と一致し、ベジエ曲線と同等になる

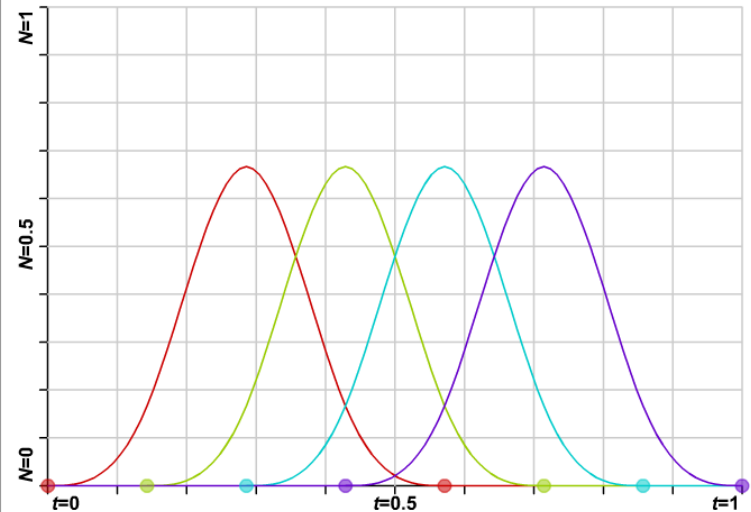
# B スプライン曲線まとめ

- 複数のセグメントを接続して1本にしたもの
  - ノット列が重要な役割をする
  - 曲線の制御が局所的
  - 接続の問題がない（常に滑らか）
  - ベジエ曲線を表現できる
- 
- 放物線以外の2次曲線を厳密に表現できない  
（NUBRBS 曲線なら対応できる）

## nurbs curve



## basis functions of points



knot vector: {0,0.14,0.29,0.43,0.57,0.71,0.86,1}

legend:  $N_0^3$   $N_1^3$   $N_2^3$   $N_3^3$

## preset examples

**bézier curves demo:** linear quadratic cubic A cubic B cubic C

**rational bézier demo:** parabola hyperbola ellipse full ellipse

**b-spline degree demo:** linear quadratic quadratic non uniform cubic

**knot vector demo:** uniform non uniform A non uniform B non uniform C

**moving one control pt:** positions A positions B

**closed b-splines:** clamped overlap unclamped

**circles as NURBS:** seven points nine points 13 points

## example description

this example shows a cubic b-spline - because number of control points is degree+1, there is no internal knot and the curve is a bézier cubic curve  
try to increase number of control points, change degree of curve or another example

## basic settings

**degree of the curve:** 3

**number of control points:** 4

**NURBS curve:**  allow weights

reset weights to 1.0

**reset knot vector:**

to clamped uniform

to unclamped uniform

**make a copy of the curve:**

new duplicate

delete duplicate

## help

change degree, number of control points and other basic settings above  
change knot vector by dragging knot at  $t$  axis of basis functions  
change control points position by dragging them  
allow weights of control points (rational curve) and change next to its position

## credits

nurbs demo ... v.1.0.1.en ... 2010-12 © Jan Foretník ... inspired by wolfram demo

# 曲線の種類

- パラメトリックな自由曲線

補間方式                      スプライン補間曲線

制御点方式                    ベジエ曲線、B スプライン曲線

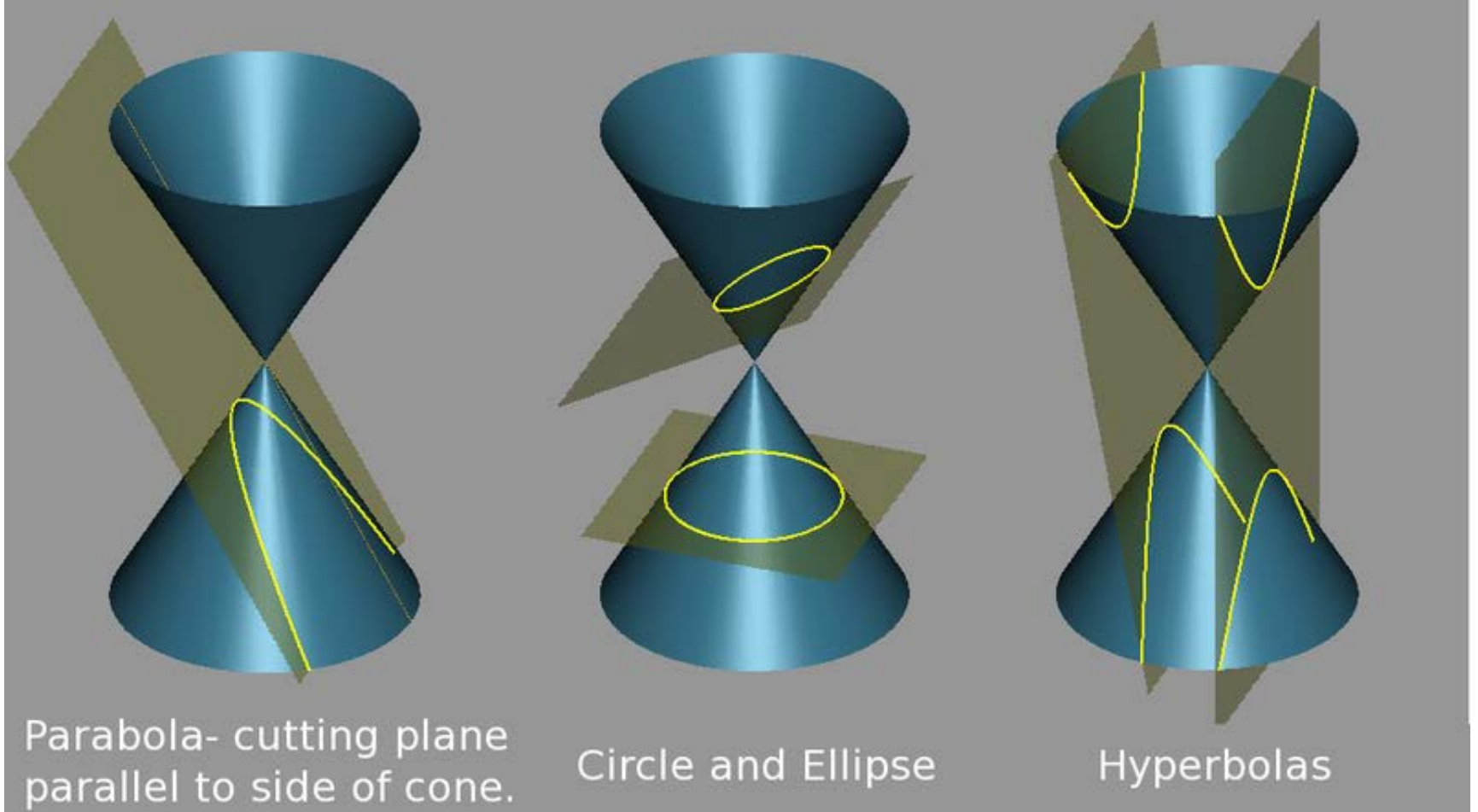
- 2 次曲線 (円錐曲線)

円、楕円、放物線、双曲線、 (直線)

円錐の切断によって得られる  $x, y$  の 2 次式



# 2次曲線 (円錐曲線)



(放物線)

(円と楕円)

(双曲線)

# 円錐曲線とパラメトリック曲線

- ベジエ曲線や B スプラインで統一的に表現できるか？

- 放物線は可能 (パラメータ  $t$  に関する 2 次式の曲線)

$$\begin{cases} x = t \\ y = a t^2 + b t + c \end{cases}$$

- 円や楕円、双曲線は**不可**

楕円 ( $a = b$  なら円)

$$\begin{cases} x = a \cos t \\ y = b \sin t \end{cases}$$

双曲線

$$\begin{cases} x = a \sec t = \frac{a}{\cos t} \\ y = b \tan t \end{cases}$$

- 円や楕円、双曲線は無限回微分可能だが、ベジエ曲線や B スプラインは有限回しか微分可能ではない

# 有理ベジエ曲線

- ベジエ曲線の式を有理式に拡張したもの
  - 有理式なら無限回微分可能
- 制御点ごとに、重み  $w_i$  をかけ合わせる
- 放物線以外の2次曲線（楕円や双曲線）も厳密に表現できる

$$P(t) = \frac{\sum_{i=0}^n w_i P_i B_i^n(t)}{\sum_{i=0}^n w_i B_i^n(t)}$$

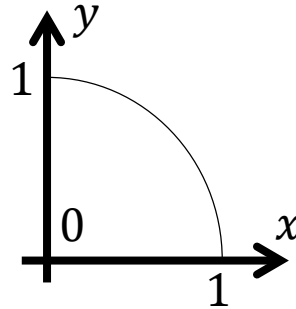
# 参考：円の有理表現

- 円の角度のパラメータ  $\theta$  を使って  $t = \tan \frac{\theta}{2}$  とおくと

$$\cos^2 \frac{\theta}{2} + \sin^2 \frac{\theta}{2} = 1 \quad \text{より、} \quad \cos^2 \frac{\theta}{2} = \frac{1}{1+t^2}$$

- この関係式から円弧を次のようにパラメータ化できる
  - $t$  が 0 から 1 まで動くとき  $\frac{1}{4}$  の円弧（単位円）になる

$$\begin{cases} x = \frac{1-t^2}{1+t^2} \\ y = \frac{2t}{1+t^2} \end{cases}$$



- 2次有理ベジエ曲線で厳密に表現可能
  - 重みは例えば  $w_0 = w_1 = 1, w_2 = 2$
  - 制御点は例えば  $P_0 = (1, 0), P_1 = (1, 1), P_2 = (0, 1)$

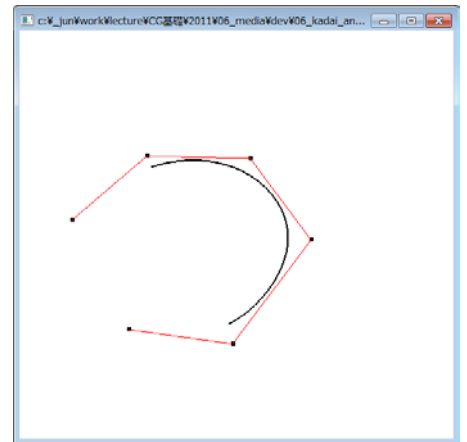
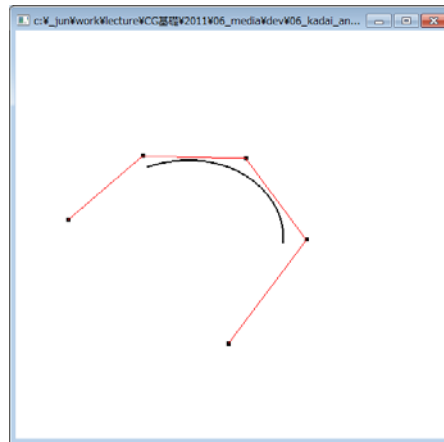
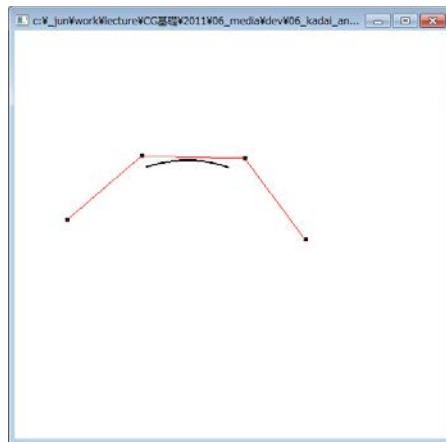
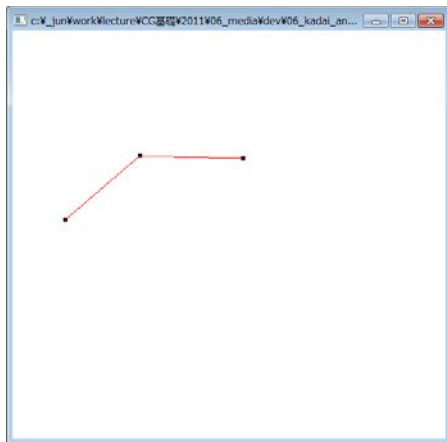
# NURBS 曲線

- NURBS : Non-Uniform Rational B-Spline  
(非一様有理 B スプライン)
- B-Spline 曲線の式を有理式に拡張したもの
- 制御点ごとに、重み  $w_i$  をかけ合わせる  
(制御点、ノット列、重みの組み合わせで  
形が決まる)
- 放物線以外の 2 次曲線 (楕円や双曲線) も厳密に  
表現できる

$$P(t) = \frac{\sum_{i=0}^{n+L-1} w_i P_i N_i^n(t)}{\sum_{i=0}^{n+L-1} w_i N_i^n(t)}$$

# 課題

# B スプライン曲線を描画する



デモ

課題を上手にこなすためのヒント

TA を有効活用する



TAのリソース

$75\text{分} \times 3\text{人} = 225\text{分人}$

受講生の人数（3C113に約50人）

$225 \div 50 = 4.5\text{分}$

一人当たり5分弱は個別指導してもらう権利がある

隣の友達と2人で質問すれば10分！