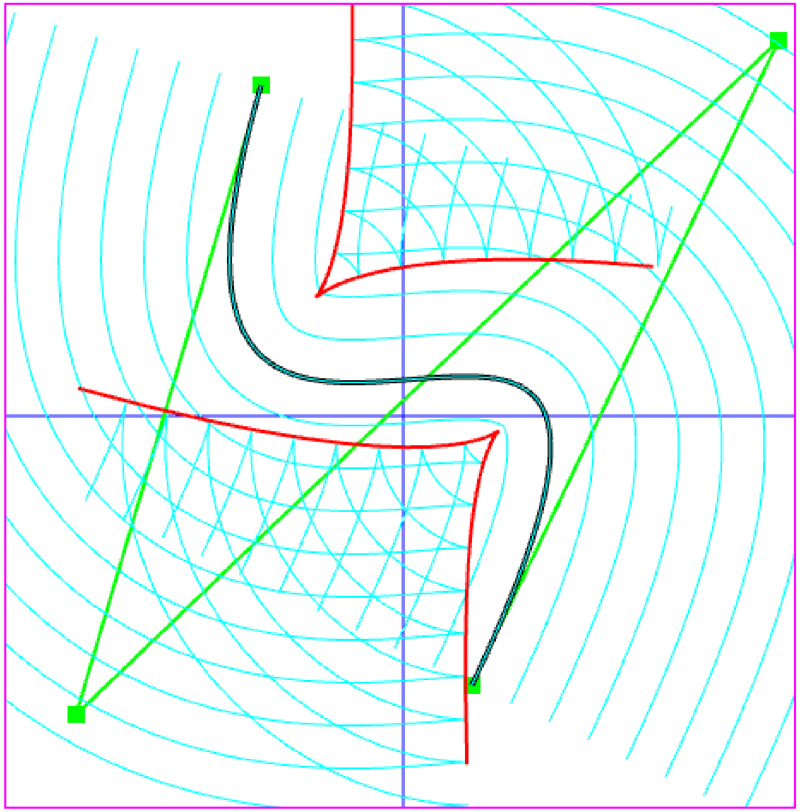
Bézier and B-spline curves for CAD

Kenjiro T. Miura
Shizuoka University

Based on Prof. Ohtake's ppt ：
http://www.den.rcast.u-tokyo.ac.jp/~yu-ohtake/GeomPro/

Objectives

- Learn basics of the curves frequently used in CG and CAD.
- Learn offset and curvature of the parametric curve.

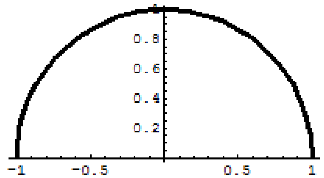


Outlines

- What is a parametric curve？
- Bézier curve
- Tangent and normal
- Curvature
- B-spline curve
- Space curve

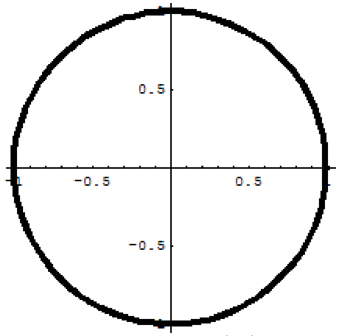
Representation of plane curve

- Explicit : $y = f(x)$
  - Limited d.o.f.



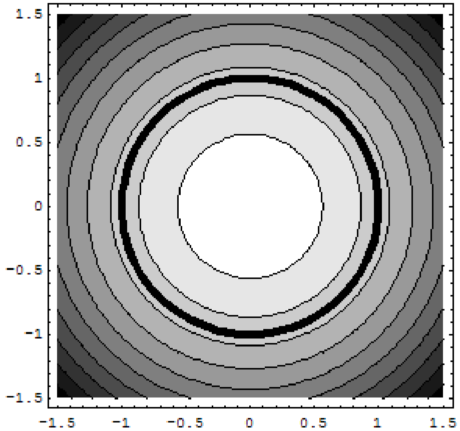
$$y = \sqrt{1 - x^2}$$

- Parametric : $\mathbf{c}(t) = \big(x(t), y(t)\big)$
  - Easy to use, high d.o.f.
  - Easy to extend to space curves



$$\mathbf{c}(t) = (\cos t, \sin t)$$

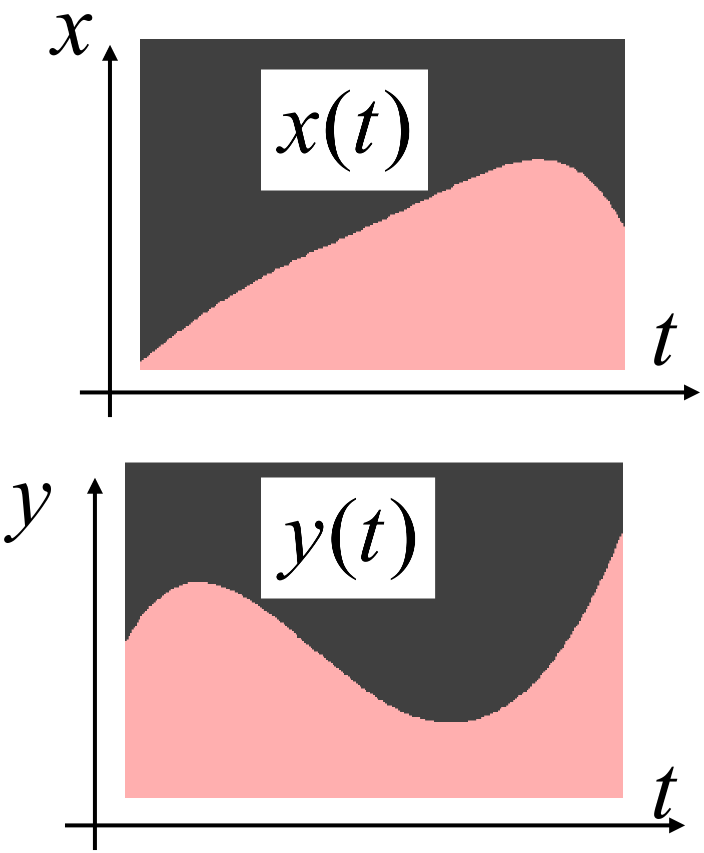
- Implicit : $f(x, y) = 0$
  - Also called iso-lines



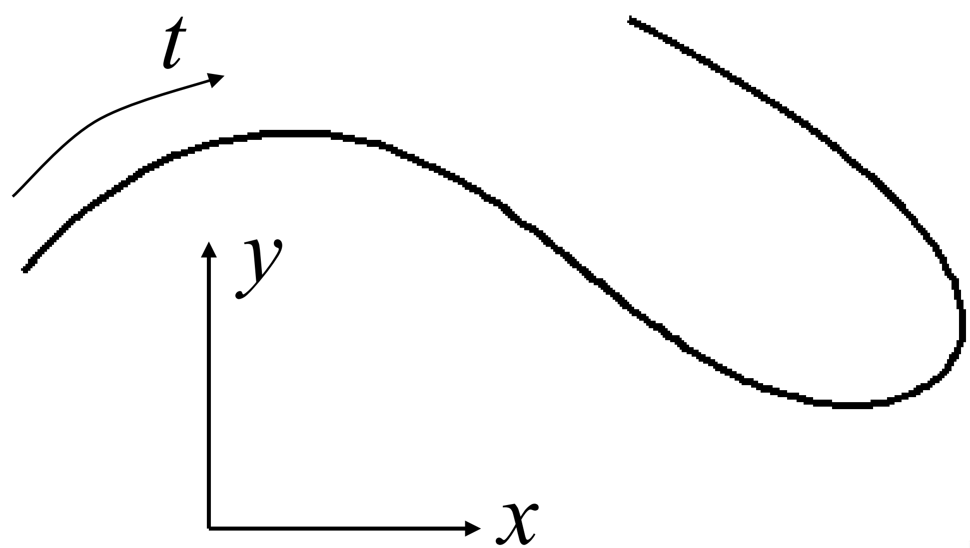
$$1 - x^2 - y^2 = 0$$

Parametric

- Explicit representation for each coordinate



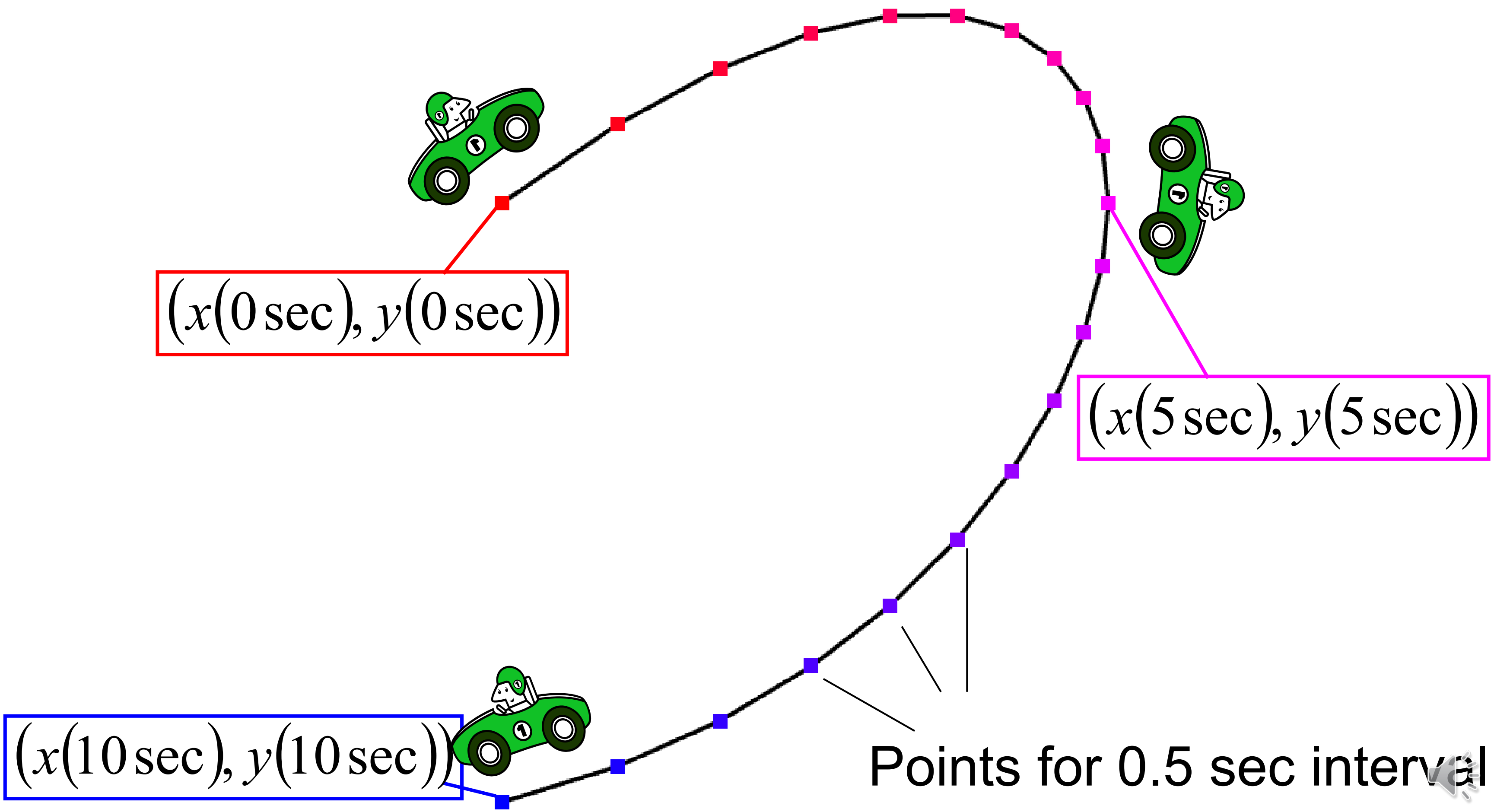
$$\mathbf{c}(t) = \big(x(t), y(t)\big)$$



Rendering of parametric curve

- Use polyline

$$\left(x(t), y(t)\right), \quad 0\,\text{sec} \le t \le 10\,\text{sec}$$

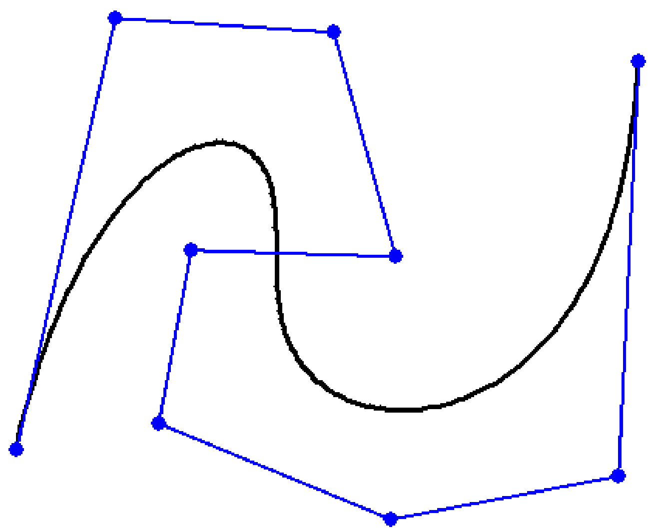


Outlines

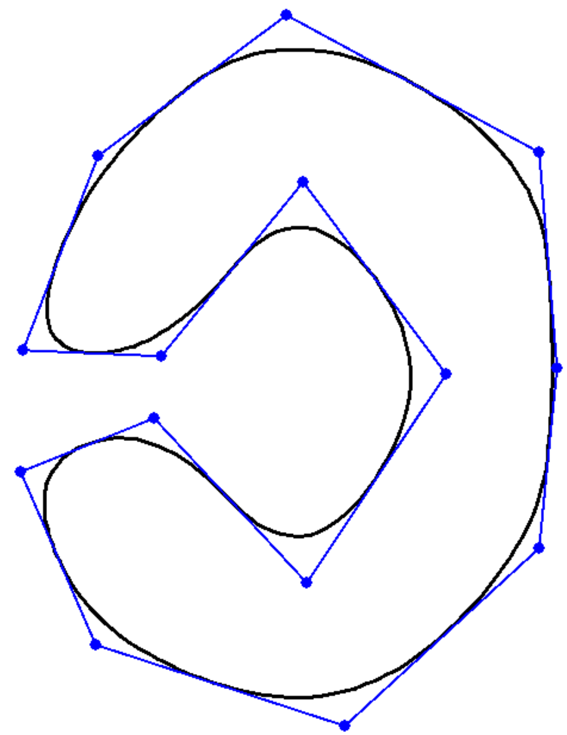
- What is a parametric curve？
- Bézier curve ←
- Tangent and normal
- Curvature
- B-spline curve
- Space curve

Frequently used for CAD

- Bézier curve
- B-spline curve



Bézier curve



B-spline closed curve

Practical example

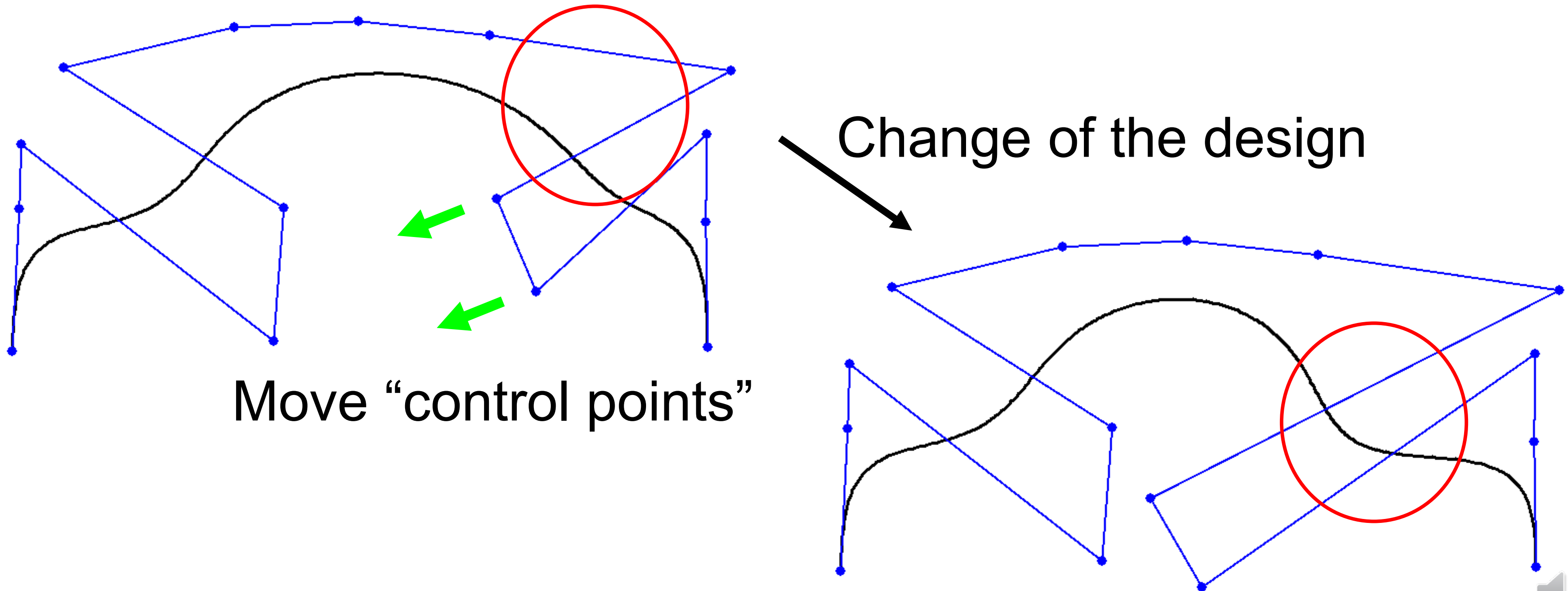
- Outline font
  - Always smooth with any zooming

Dot font
(raster imags)

Outline font
(vector image)

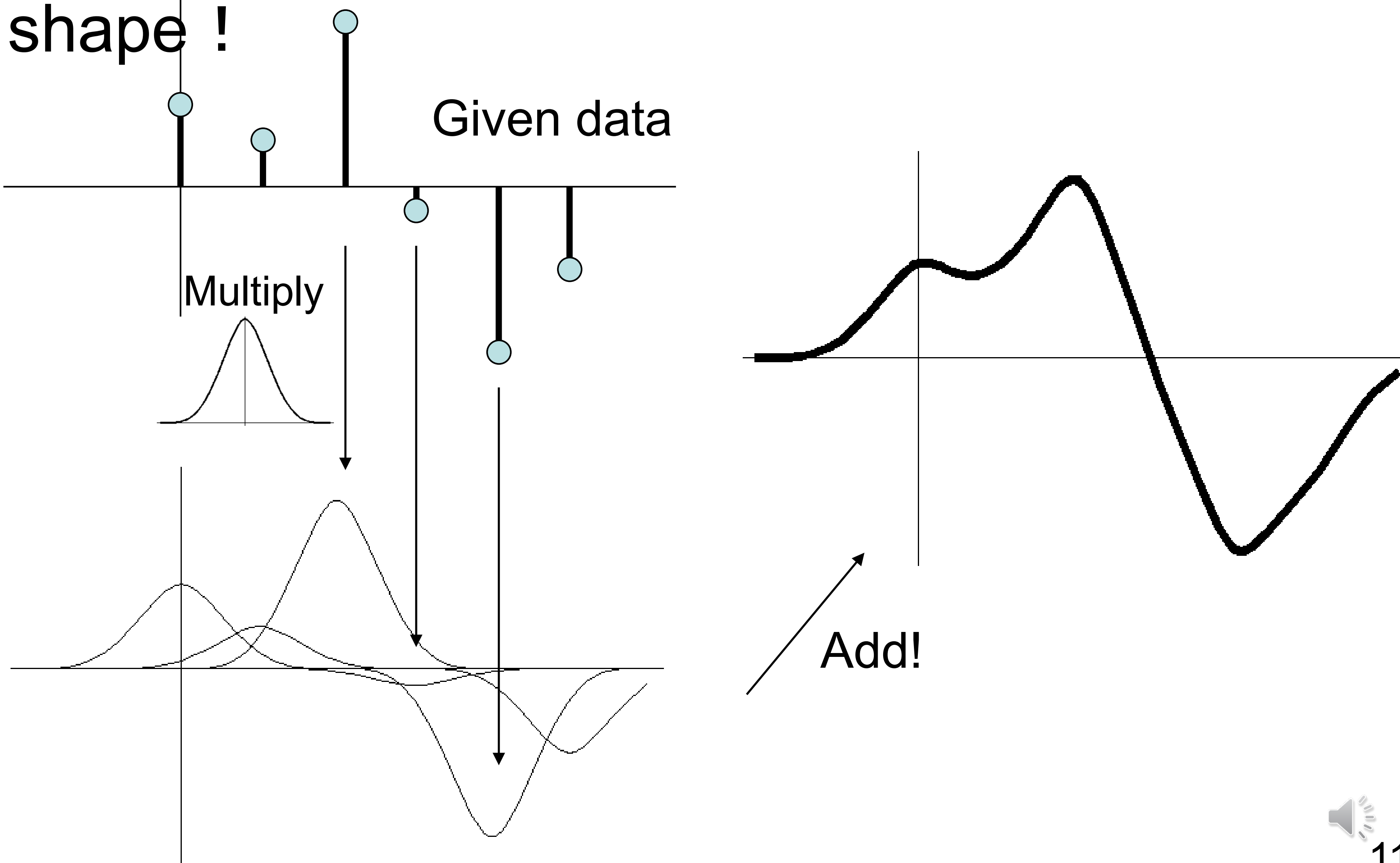
Problem dealt by Bézier
(For car design)

- Generate a curve based on a sequence of points in a plane.
- Change the shape according to the move of the points



Bézier's idea

- Add smooth functions to represent the whole shape !



Bernstein basis function

- Probability of “Repeated trial”
  - Probability for t trial, n times, get i times.

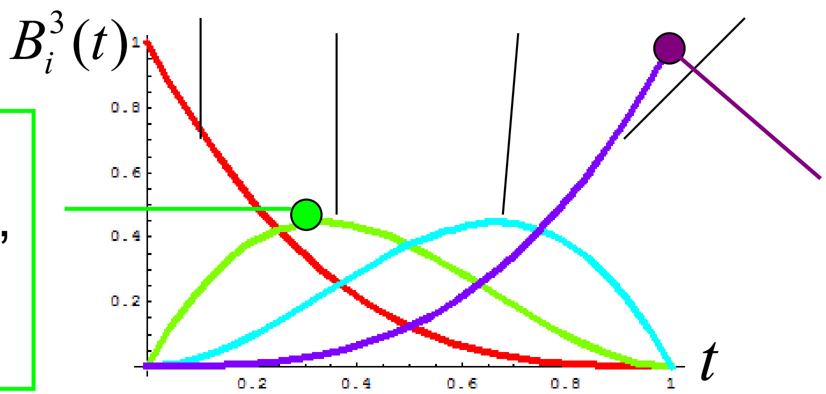
$$ {}_{n}C_{i}\, t^{i} (1-t)^{n-i} $$

↓

$$ B_i^n(t) $$ と書く

$0 < t < 1$

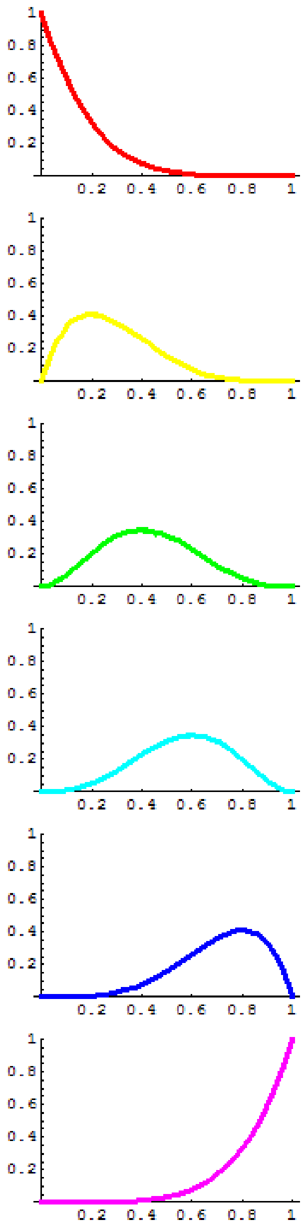
$$ \{B_0^3(t), B_1^3(t), B_2^3(t), B_3^3(t)\} $$



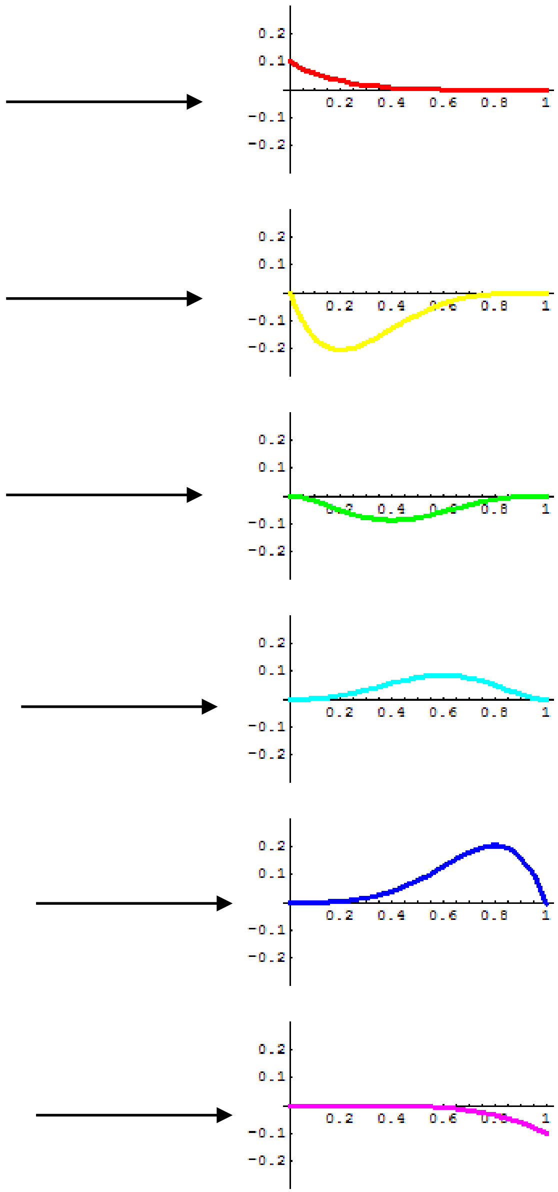
A lot : win a time among three tries, Win one time.

100% lot
try three times
Three wins: 100%

$B_i^n(t)$

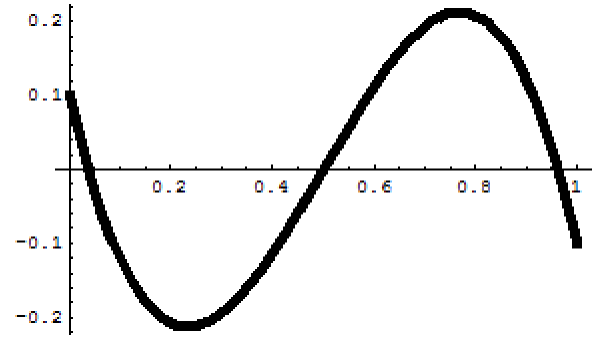


$B_i^n(t)\,x_i$



$$\sum_{i=0}^{n}$$

$$x(t) = \sum_{i=0}^{n} B_i^n(t)\,x_i$$

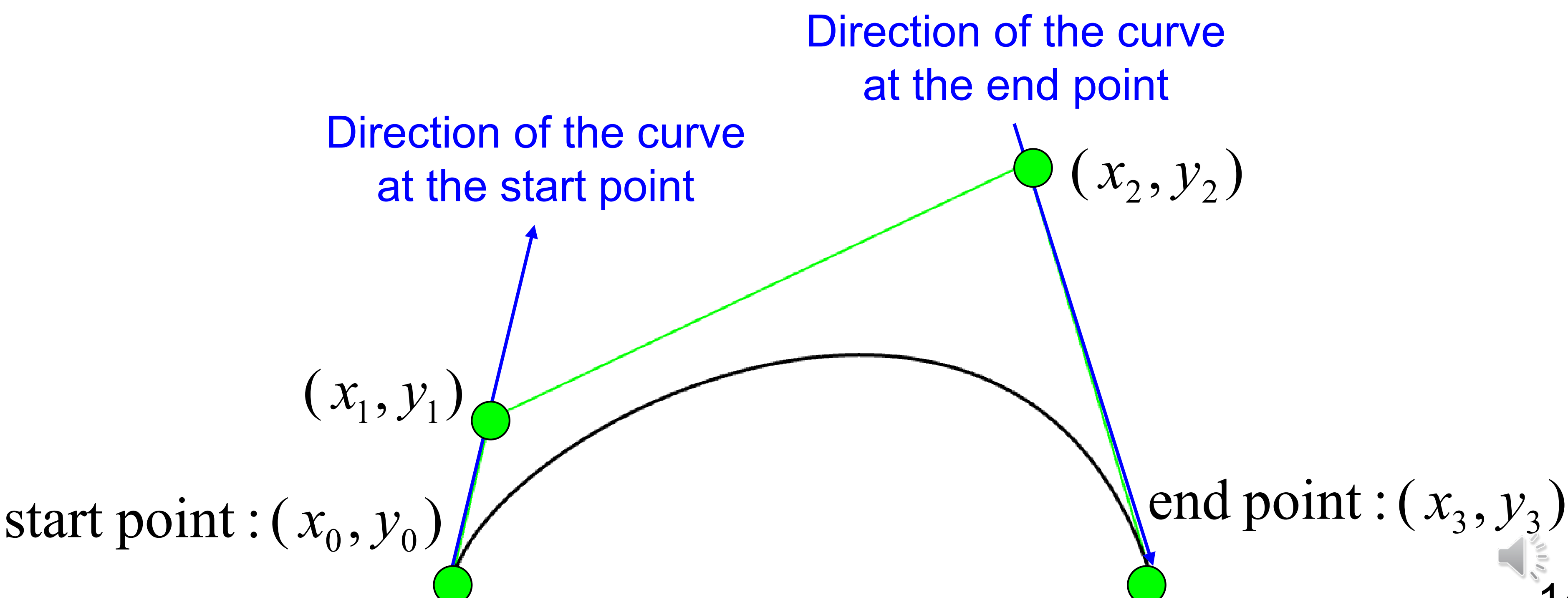


Same for y

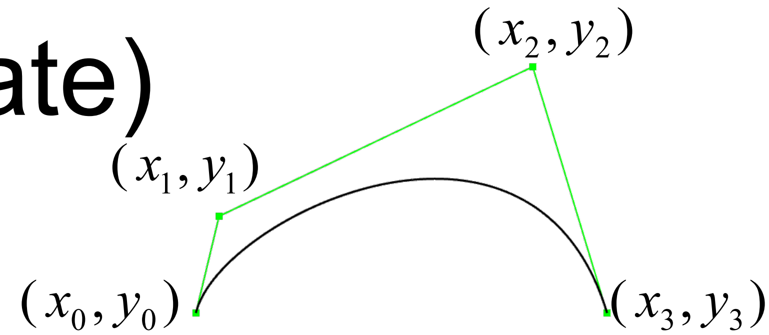
$$\begin{cases} x(t) = \sum_{i=0}^{n} B_i^n(t)\,x_i \\ y(t) = \sum_{i=0}^{n} B_i^n(t)\,y_i \end{cases}$$

Cubic Bézier curve

- The most frequently used curve
  - # of control points = 4
  - Convenient for spedifying the start and end points and tangent vectors there.
  - Cubic polynomials → Next page



Cubic Bézier curve (x coordinate)



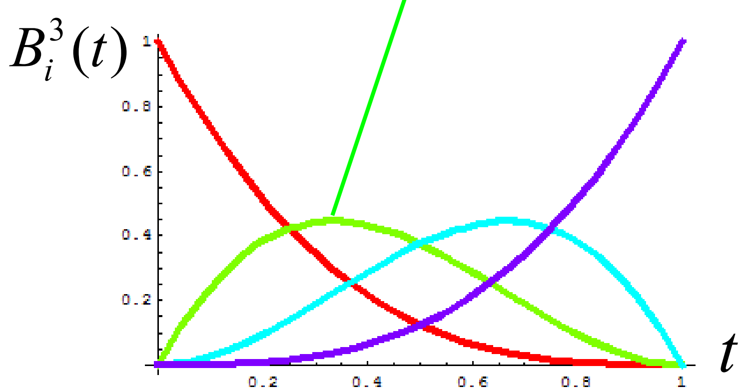
$$x(t) = \sum_{i=0}^{3} B_i^3(t)\, x_i$$

$$= B_0^3(t)\, x_0 + B_1^3(t) x_1 + B_2^3(t)\, x_2 + B_3^3(t)\, x_3$$

$$B_1^3(t) = {}_3C_1 t^1 (1-t)^{3-1}$$
$$= 3t(1-t)^2$$

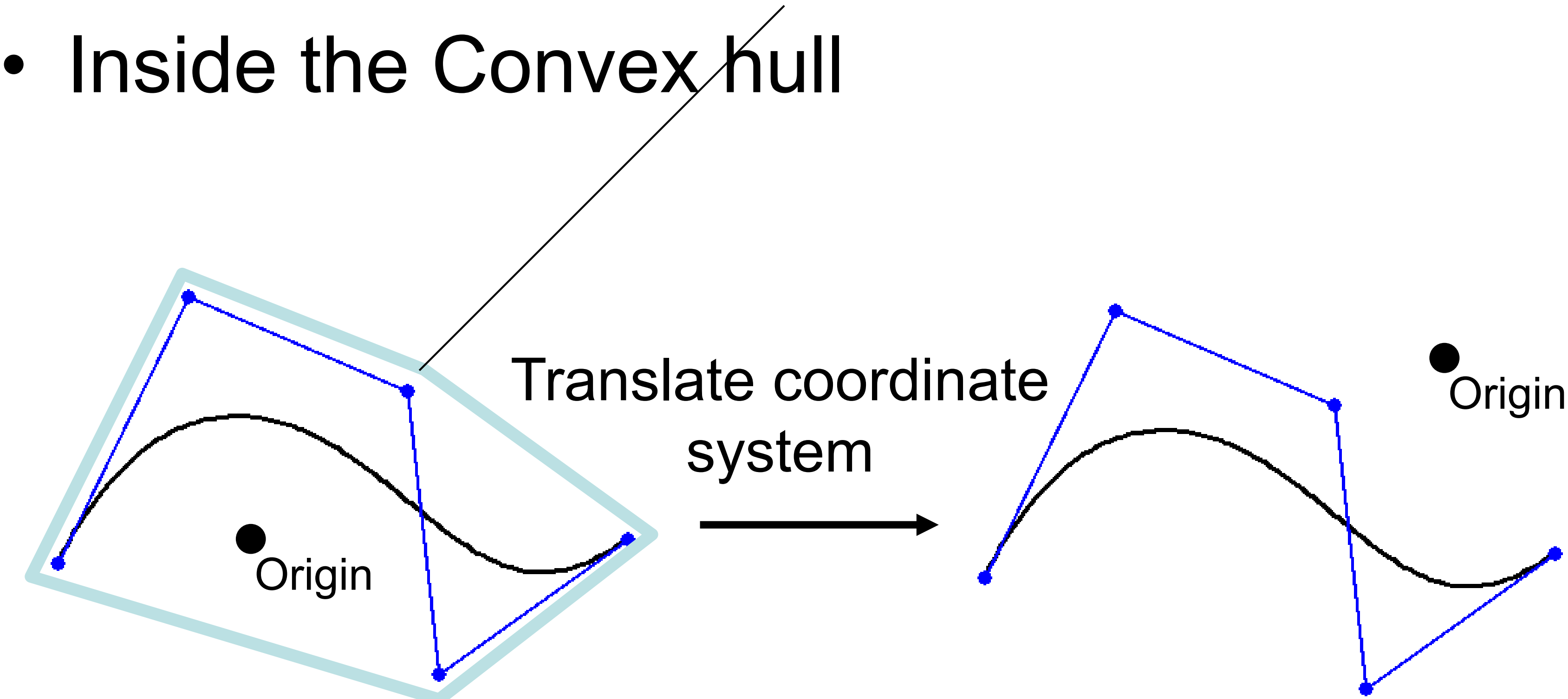
cubic polynomial of t

- $t = 0$, 0
- $t = 1$, double roots
- $t = 1/3$, max



Properties of Bézier curve

- Remains the same shape with a coordinate system translation.
- Inside the Convex hull



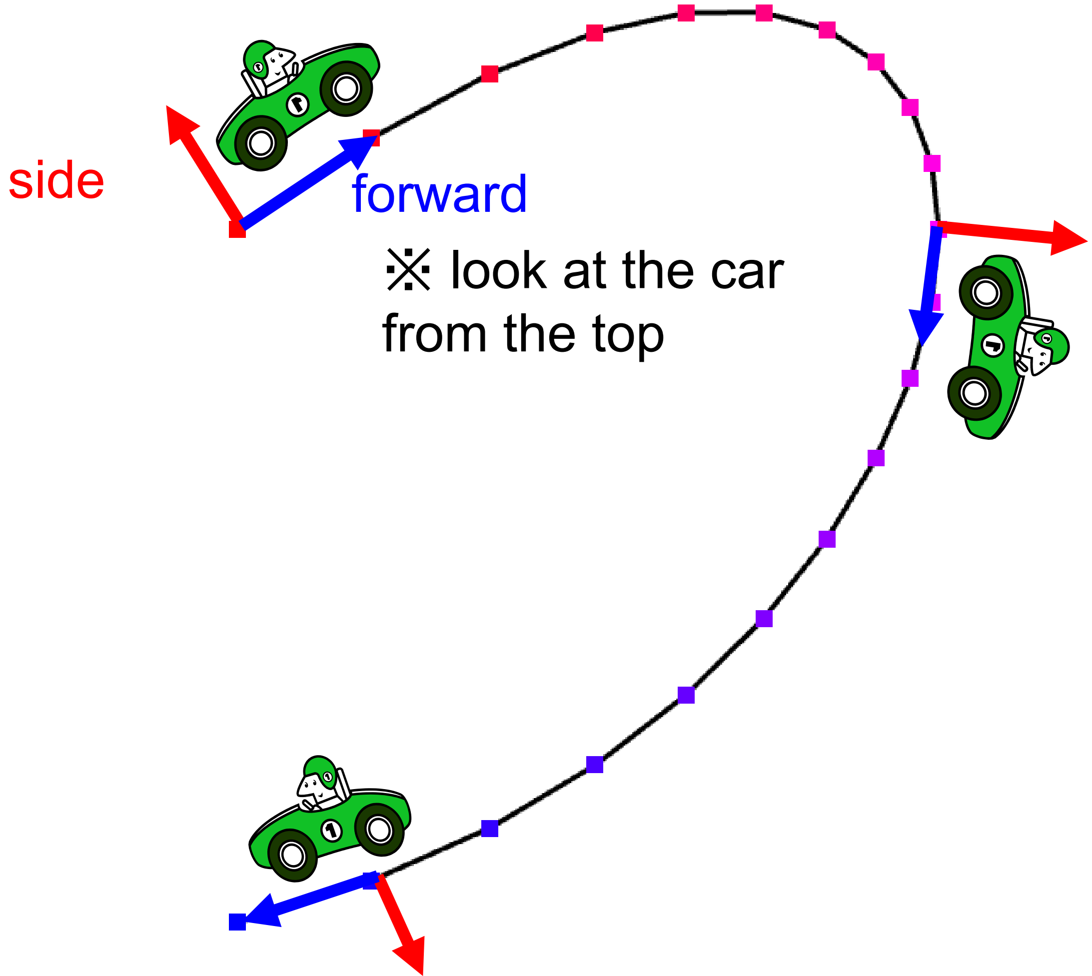
Reason : point on the curve is a positive weighted sum of the control points

Outlines

- What is a parametric curve？
- Bézier curve
- Tangent and normal ←
- Curvature
- B-spline curve
- Space curve

Tangent and normal vectors

- Tangent vector : direction of the curve
- Normal vector : perpendicular to tangent



Tangent vector of the parametric curve

- Differentiate with respect to the parameter

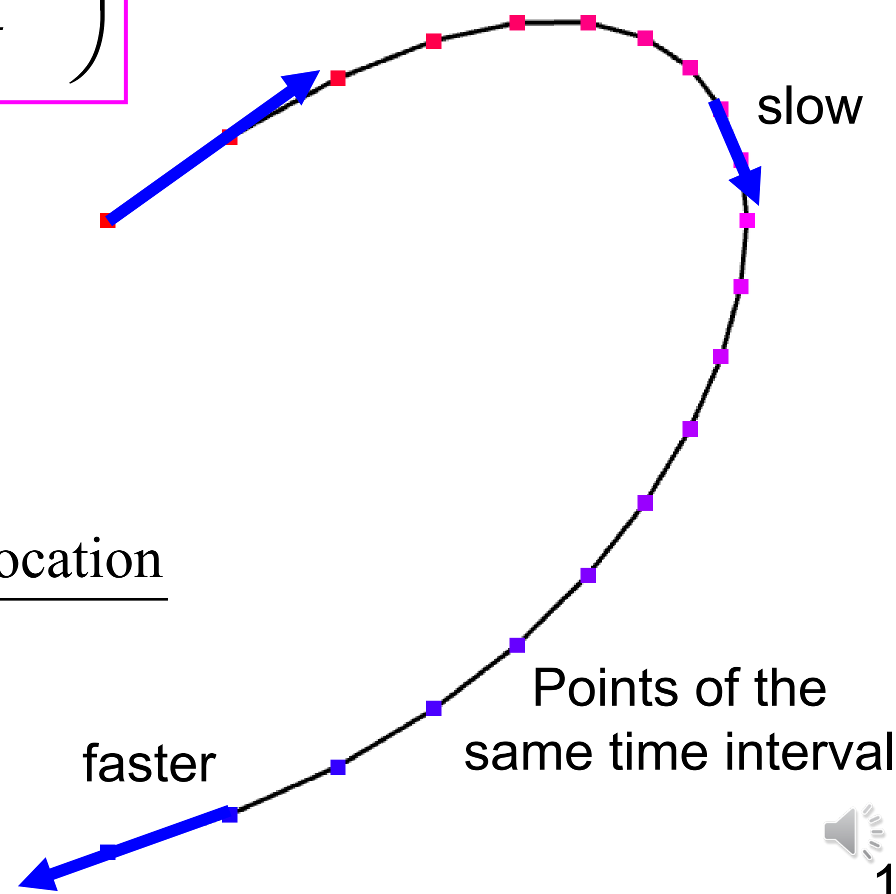
$$\mathbf{t}(t) = \left(\frac{dx(t)}{dt}, \frac{dy(t)}{dt} \right)$$

If the parameter is time,
velocity vector

$$\text{velocity vector} = \frac{\text{displacementvector}}{\text{elapsed time}}$$

$$= \frac{\text{Next location} - \text{current location}}{\text{elapsed time}}$$

Make e-time to infinitesimal
= differentiate



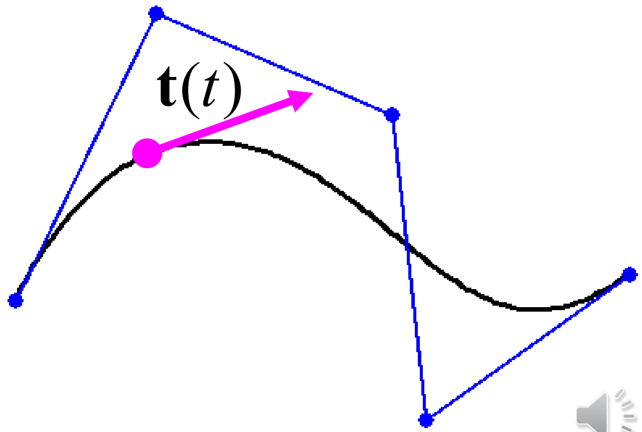
Tangent vector of Bézier curve

- Differentiate the basis functions
  - Control points are coefficients

$$\begin{cases} \dfrac{dx(t)}{dt} = \dfrac{d}{dt}\displaystyle\sum_{i=0}^{n} B_i^n(t)\, x_i = \sum_{i=0}^{n} \dfrac{dB_i^n(t)}{dt} x_i \\ \dfrac{dy(t)}{dt} = \dfrac{d}{dt}\displaystyle\sum_{i=0}^{n} B_i^n(t)\, y_i = \sum_{i=0}^{n} \dfrac{dB_i^n(t)}{dt} y_i \end{cases}$$

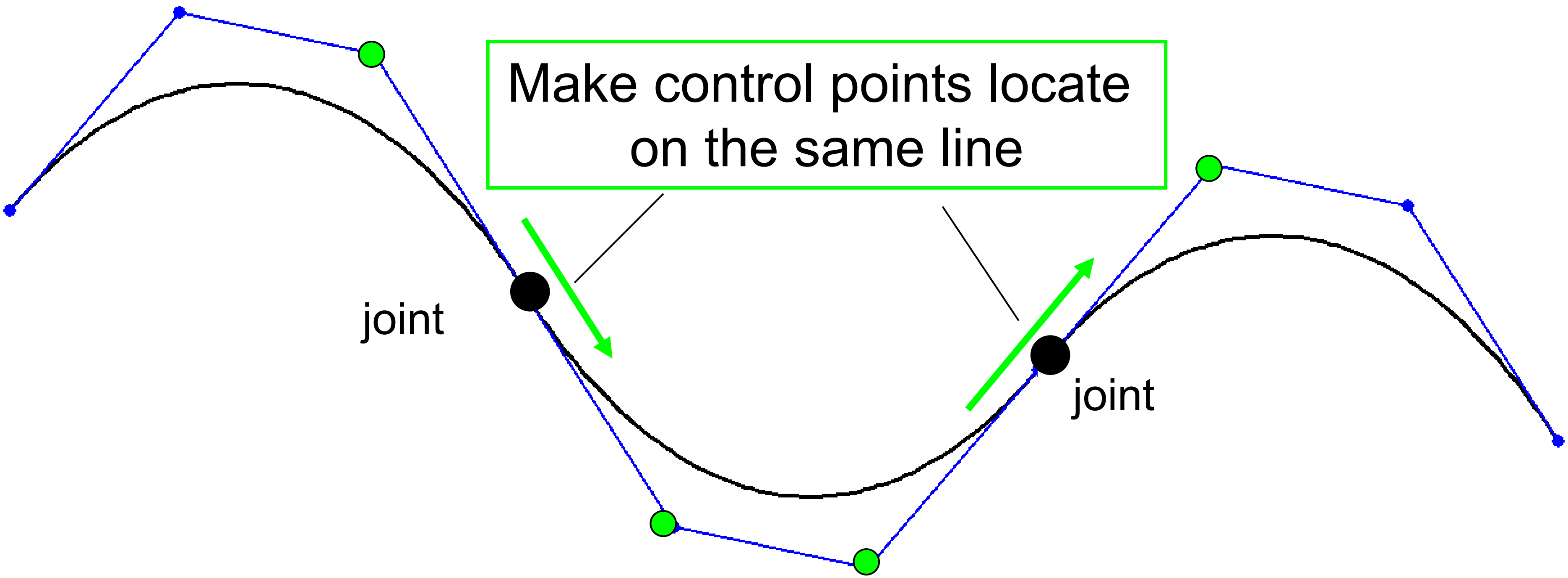
First derivative of B_1

$$\frac{dB_1^3(t)}{dt} = \frac{d}{dt} 3t(1-t)^2 = 3(1-t)(1-3t)$$



Bézier spline curve

- Connect Bézier curves of low degree (usually cubic)
  - Locate control points to be parallel at joints.



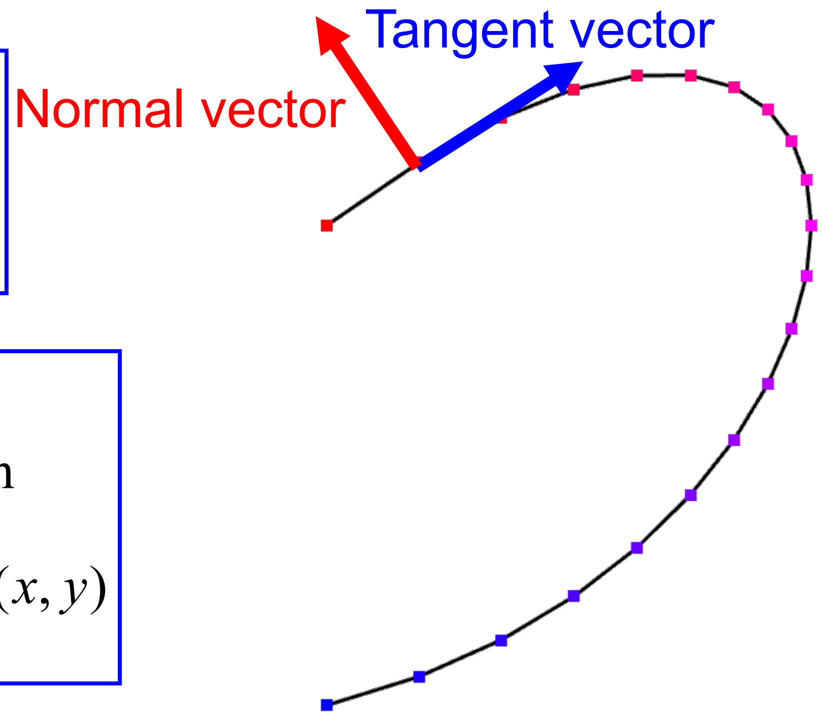
※ At the end points of a Bezier curve, tangent to the polylines made of the control points.

Normal vector

- Rotate tangent vector by 90 degrees.
  - Usually use a unit normal vector whose length=1

$$\mathbf{n}(t) = \left(-\frac{dy(t)}{dt}, \frac{dx(t)}{dt}\right) \Big/ \sqrt{\left(\frac{dx(t)}{dt}\right)^2 + \left(\frac{dy(t)}{dt}\right)^2}$$

$$\text{vector } (x, y) \quad \downarrow \text{ rotate 90 degrees anticlockwise} \quad \text{vector } (-y, x)$$



$$\text{vector } (x, y) \quad \downarrow \text{ unit length} \quad \text{vector } \frac{1}{\sqrt{x^2 + y^2}}(x, y)$$

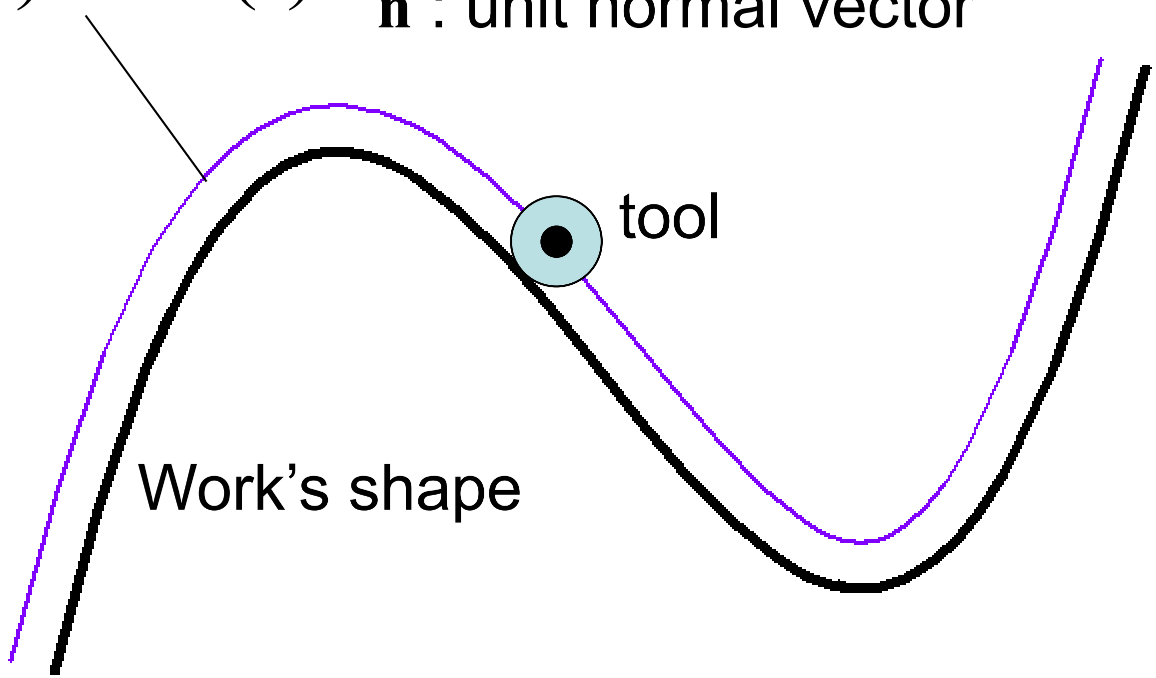
Offset curve

- Trajectory made by translating points on the curve in the direction of norma vector
  - When machine with a ball-end mill
    the path of its center is on the offset curve

$$\mathbf{c}(t) + r\,\mathbf{n}(t)$$

r : radius of the machining tool

$\mathbf{n}$: unit normal vector

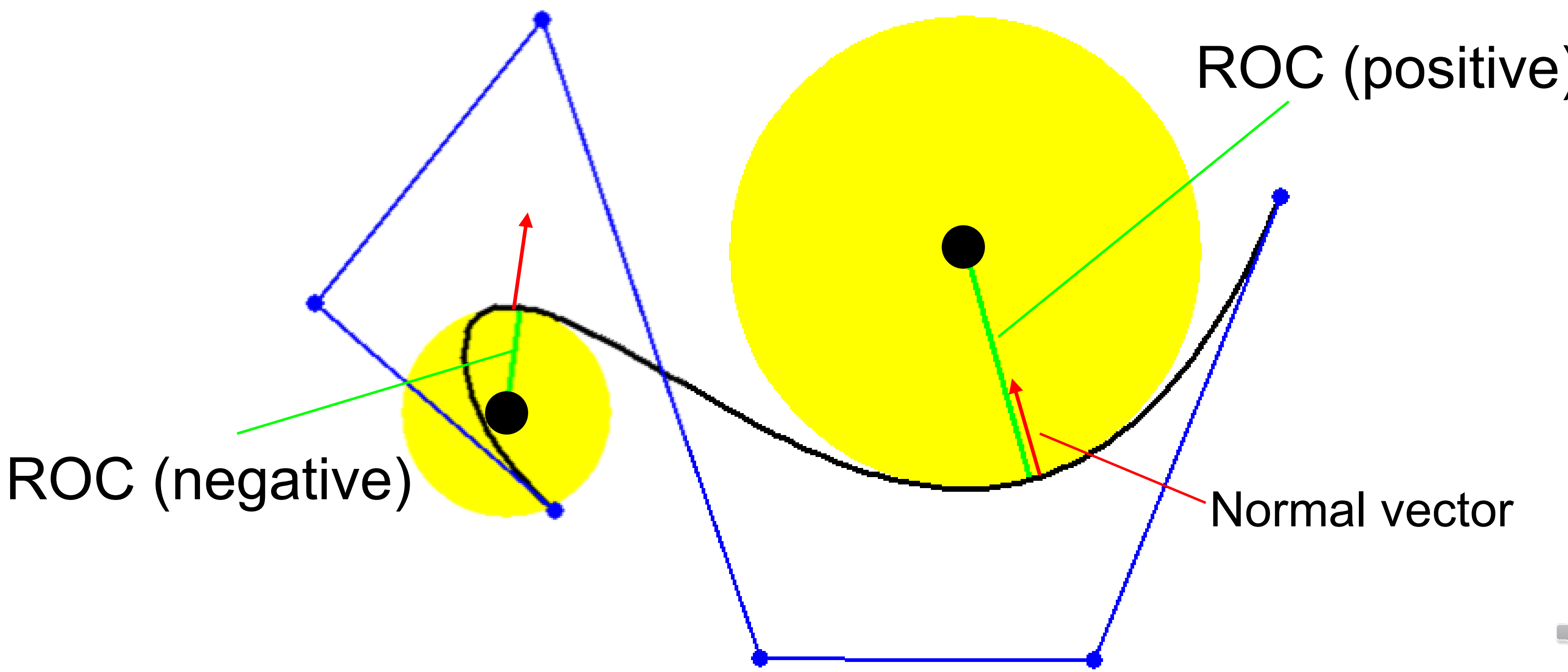


Outlines

- What is a parametric curve？
- Bezier curve
- Tangent and normal
- Curvature
- B-spline curve
- Space curve

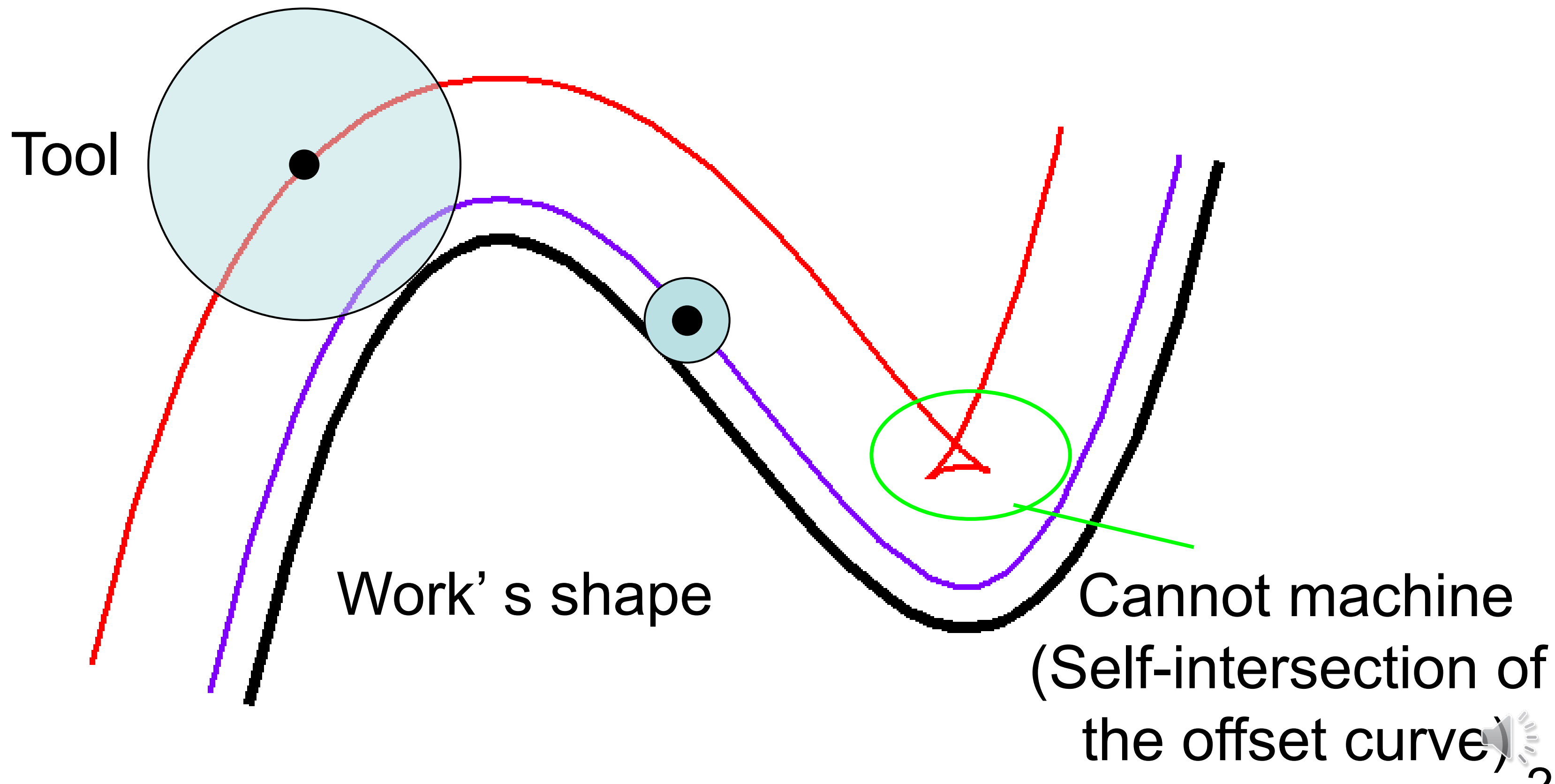
Curvature and radius of curvature

- ROC : radius of the best fit circle
  - If positive, the same direction of the normal vector
- Curvature : reciprocal of ROC (0 if straight)



Application of radius of curvature

- Cannot machine with a tool with a radius llaeger the ROC.
  - Useful for tool selection



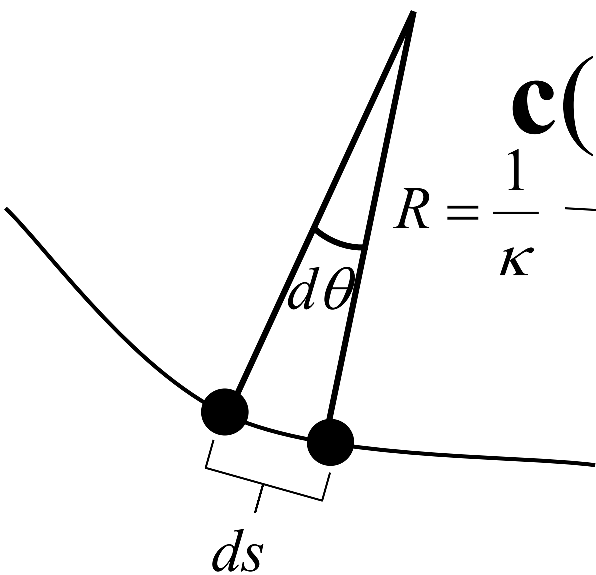
Calculation of curvature

- Approximate the curve with a circle

ROC × rotation of tangent vector [radian]
= length of the approximate arc

↓ Consider small change

$$\mathbf{c}(t) = \big(x(t), y(t)\big)$$



↓

$$\kappa = \frac{d\theta}{ds}$$

: change of the angle

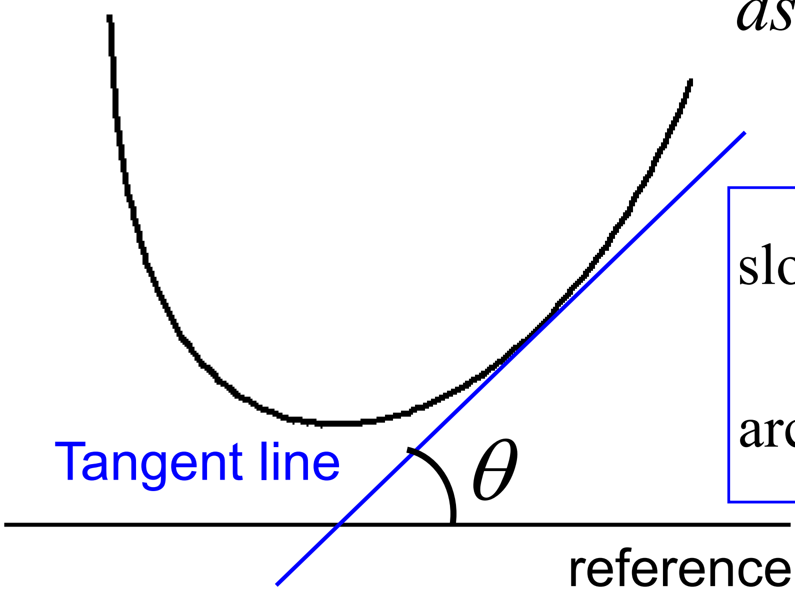
Calculation of curvature

$$\kappa(t)=\frac{\begin{vmatrix} dx(t)/dt & dy(t)/dt \\ d^2x(t)/dt^2 & d^2y(t)/dt^2 \end{vmatrix}}{\left(\left(dx(t)/dt\right)^2+\left(dy(t)/dt\right)^2\right)^{\frac{3}{2}}}$$

※ need second derivative

Use the equations below

$$\kappa(t)=\frac{d\theta(t)}{ds(t)}=\frac{d\theta(t)}{dt}\left(\frac{ds(t)}{dt}\right)^{-1}$$

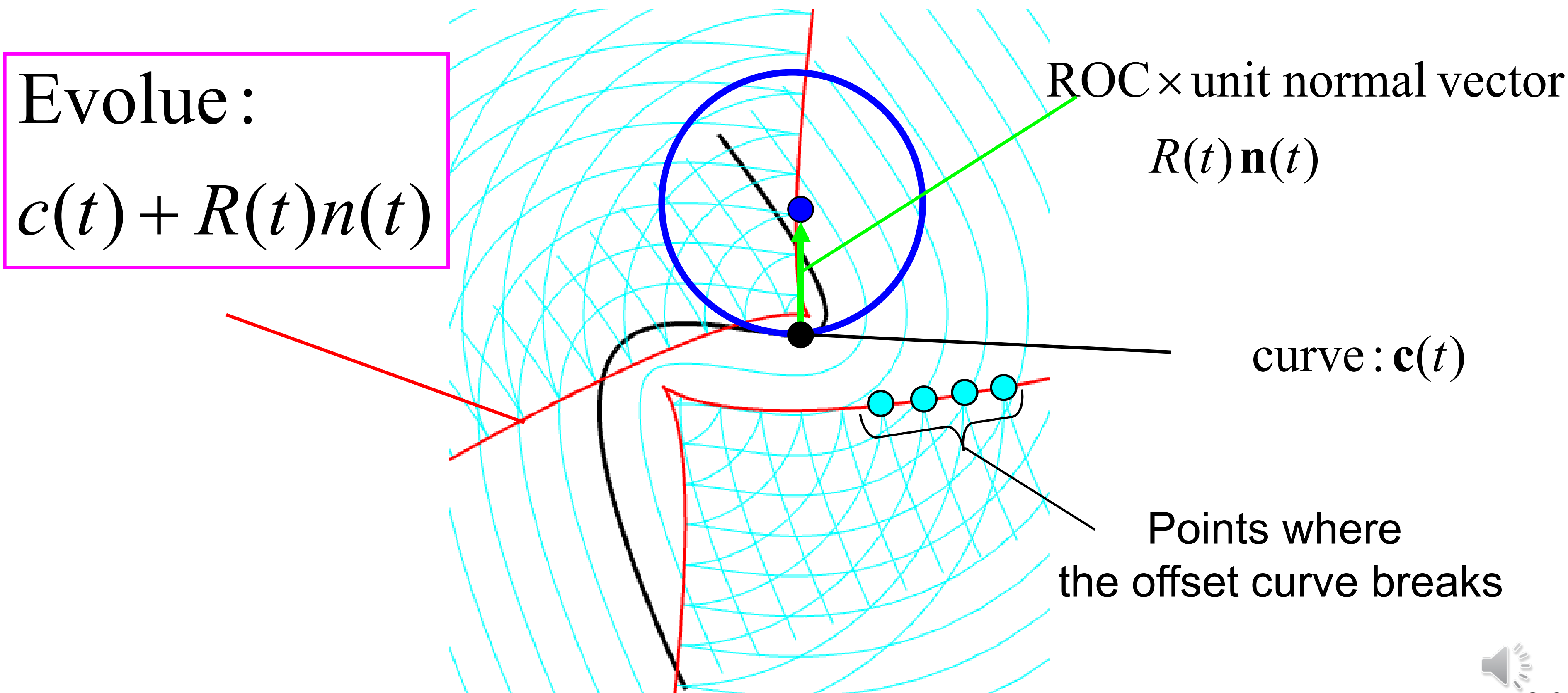


$$\text{slope of tangent line}: \tan\theta(t)=\frac{dy(t)/dt}{dx(t)/dt}$$

$$\text{arc length}: s(t)=\int_{t_0}^{t}\sqrt{\left(dx(\hat{t})/d\hat{t}\right)^2+\left(dy(\hat{t})/d\hat{t}\right)^2}\,d\hat{t}$$

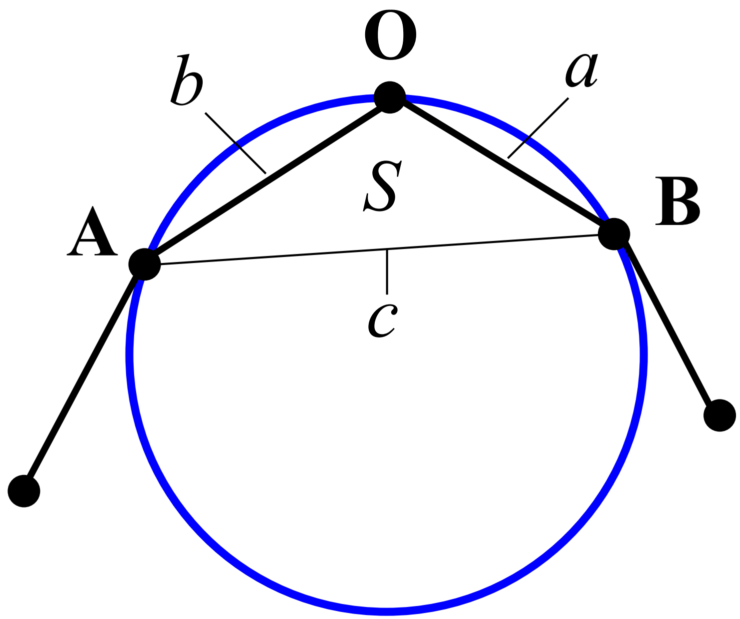
Evolute curve

- Trajectory of the best fit circle circle
  - Translate in the normal direction by the ROC.
  - Pass through points where the offset line breaks.



Approximate ROC with a polyline

- Radius of the circle which passes through 3 points
  - Remind the sine theorem



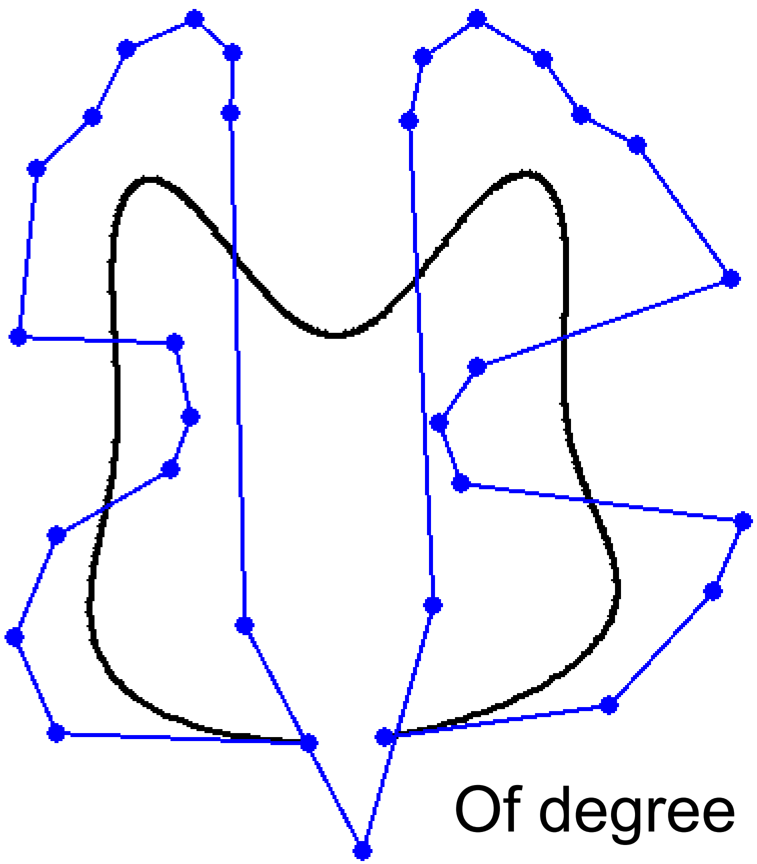
Calculate ROC
at point **O**
with the side lengths
and area of
triangle **OAB**

Outlines

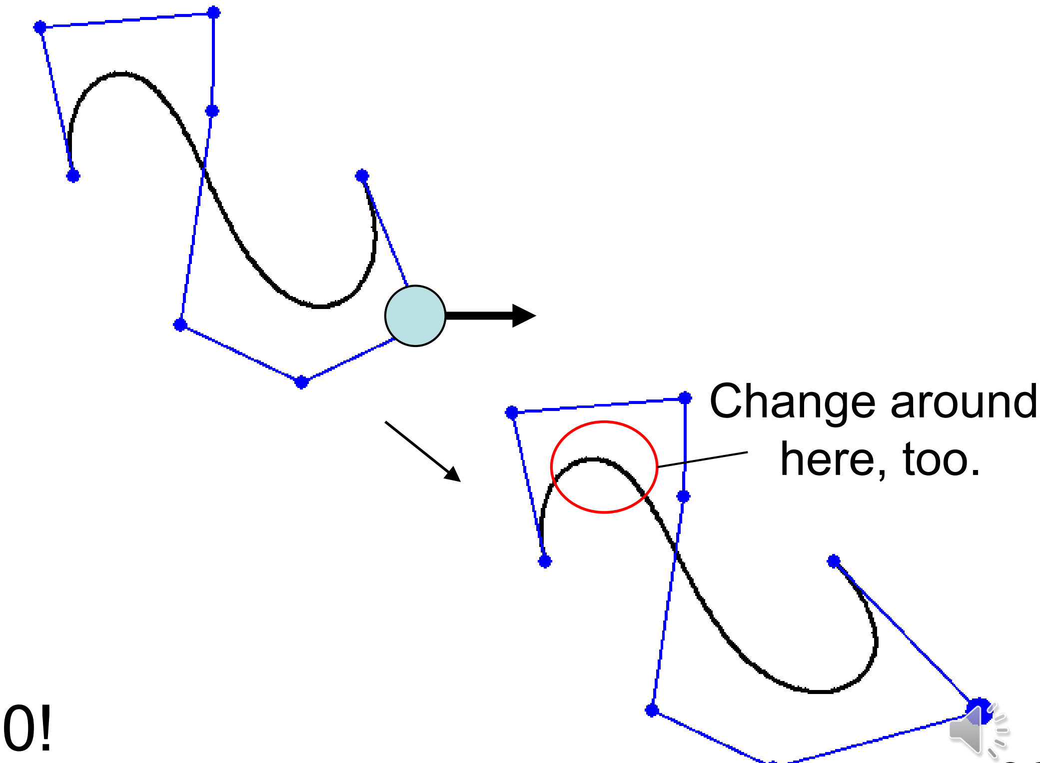
- What is a parametric curve？
- Bezier curve
- Tangent and normal
- Curvature
- B-spline curve ←
- Space curve

Problems on Bezier curve

- Degree increases with the number of control points.
- One movement will change the whole shape.

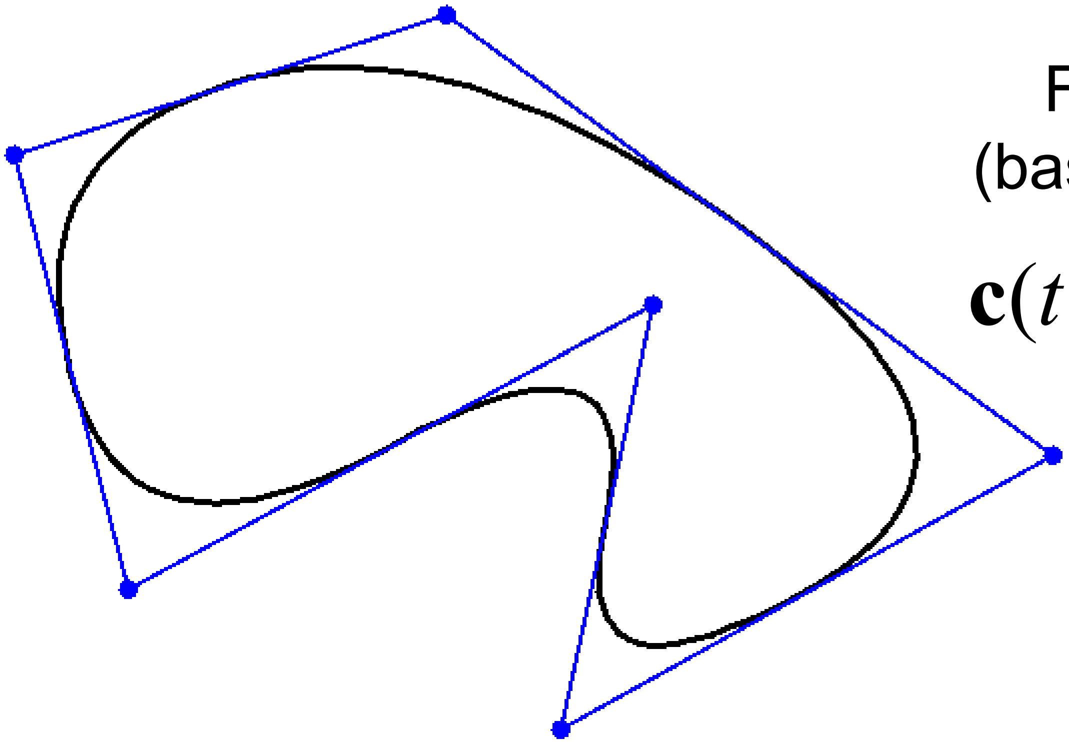


Of degree 30!



B-spline curve (different from Bézier spline)

- Example of quadratic B-spline curve
  - The curve is tangent at the middle point of the control point polyline.



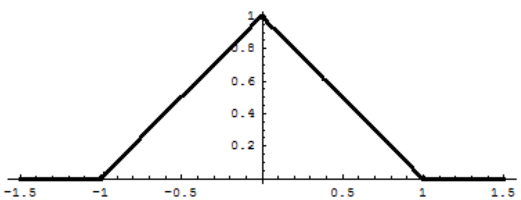
Format is the sum of
(basis func. × pos. of c.p.)

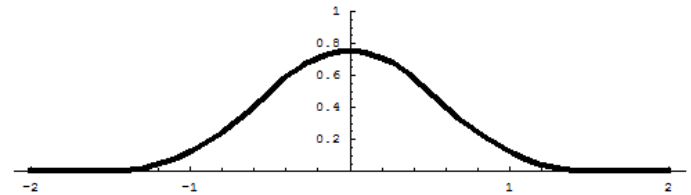
$$\mathbf{c}(t) = \sum N^2(t-i)\mathbf{p}_i$$

B-spline basis function

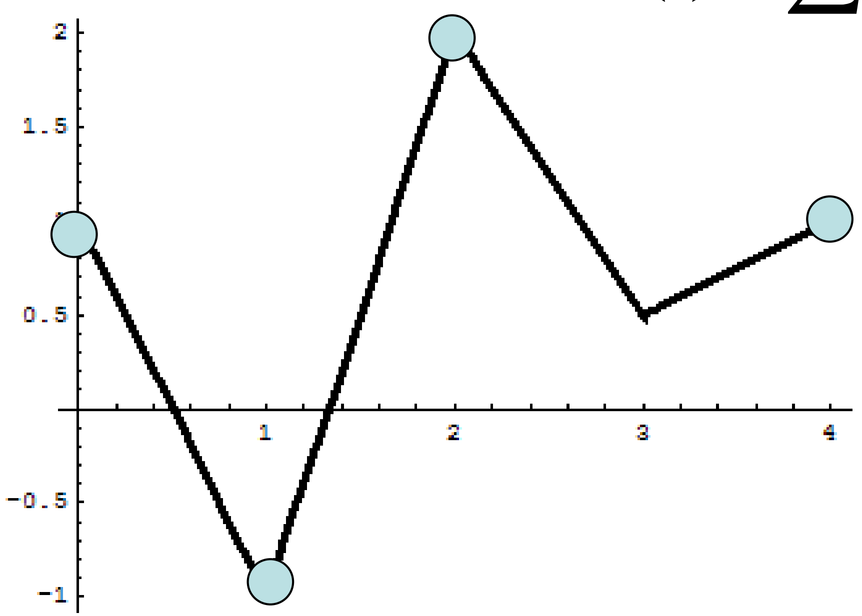
$$N^1(t) = \begin{cases} -|t|+1 & |t| \leq 1 \\ 0 & \text{otherwise} \end{cases}$$

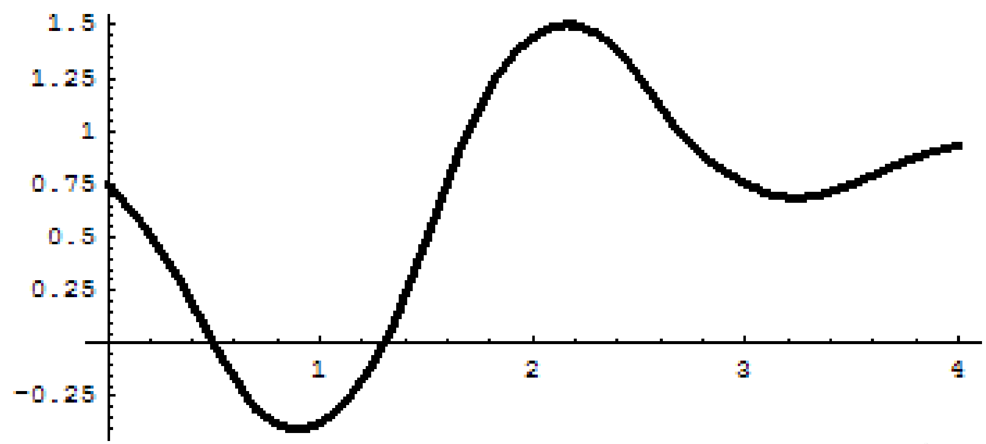
$$N^2(t) = \begin{cases} -|t|^2 + 3/4 & \text{if } |t| \leq 1/2 \\ (4|t|^2 - 12t + 9)/8 & \text{else if } |t| \leq 3/2 \\ 0 & \text{otherwise} \end{cases}$$





$$x(t) = \sum N^2(t-i)\, x_i$$



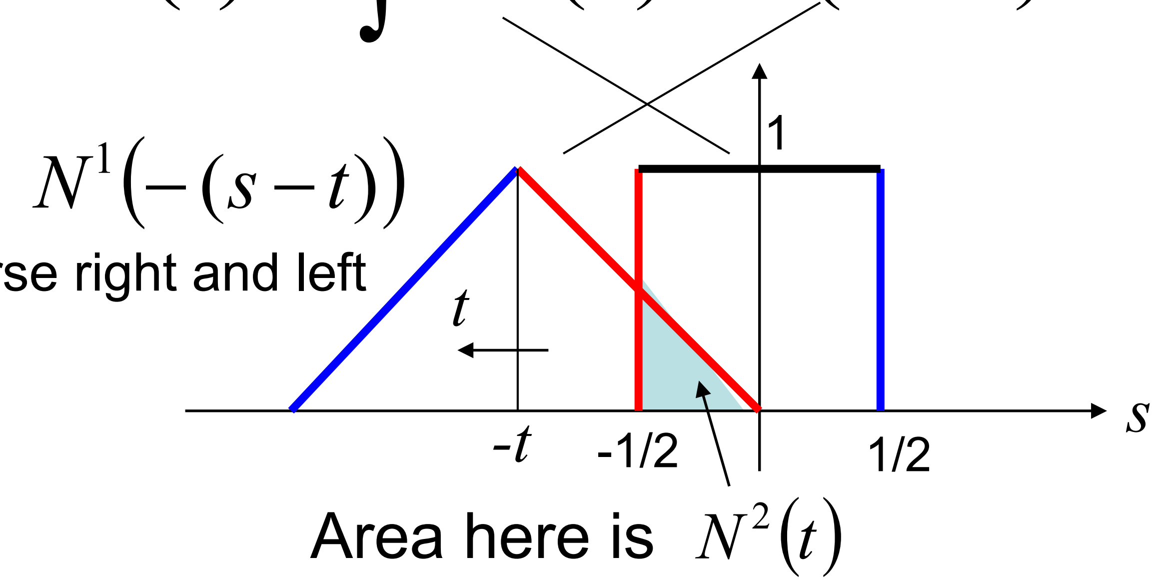


How to make B-spline basis func.

- Recursively defined by the following convolution

$$N^0(t) = \begin{cases} 1 & \text{if } |t| \le 1/2 \\ 0 & \text{otherwise} \end{cases}$$

$$N^{n+1}(t) = \int N^0(s) N^n(t-s)\,ds$$



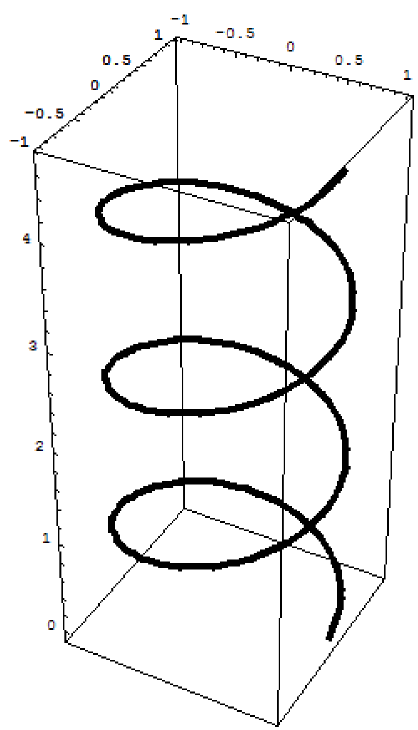
Outlines

- What is a parametric curve？
- Bezier curve
- Tangent and normal
- Curvature
- B-spline curve
- Space curve ←

Space curve in parametric form

- Only add z coordinate.

$$\mathbf{c}(t) = \big(x(t), y(t), z(t)\big)$$

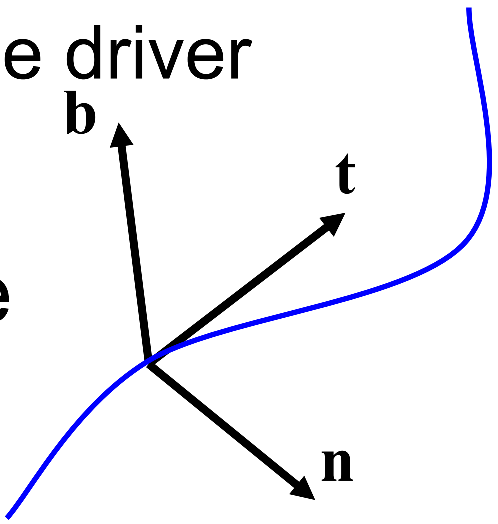


Herical curve :

$$\mathbf{c}(t) = (r\cos t, r\sin t, ht)$$

Binormal and torsion (space curve)

- Drive a car on a space curve
  - tangent **t** : ongoing direciton
  - normal **n** : curve direction(cannot defined for straight line)
  - binormal **b** : top direction for the driver



- Curvature κ : degree of curve

- Torsion τ: Vibration of head

Frenet-Serret formula (space curve)

$$\mathbf{t}'(s) = \kappa(s)\mathbf{n}(s)$$

$$\mathbf{b}'(s) = -\tau(s)\mathbf{n}(s)$$

$$\mathbf{n}'(s) = \tau(s)\mathbf{b}(s) - k(s)\mathbf{t}(s)$$