3. Interpolating Values

- Methods for precisely specifying the motion of objects

- Animator uses these techniques to directly control how the objects will move

3. Interpolating Values

Interpolation

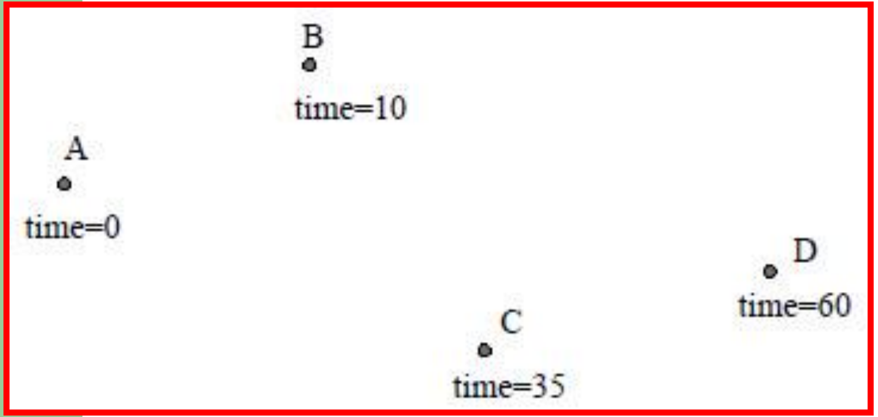
- Foundation of most animation

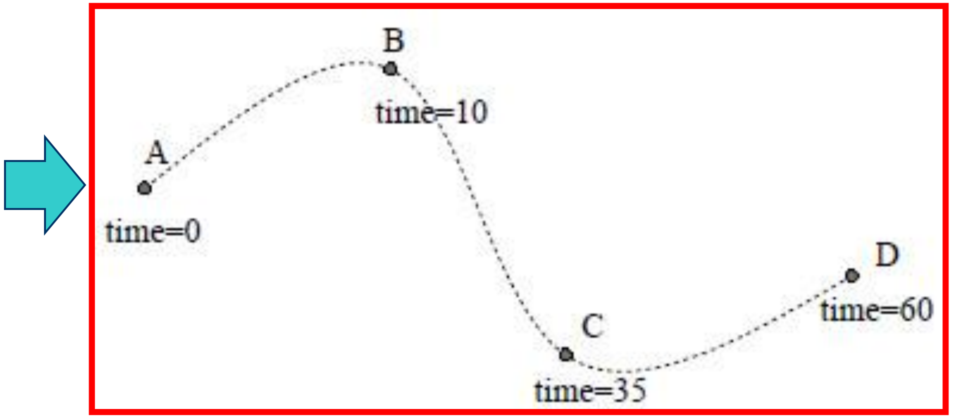
- path curve (space curve) construction
- motion control

3.1 Space Curve Construction

Space curve: path to be followed by object

- **Given:** position-time pairs constraining the motion





- **Want:** a curve interpolating these position-time pairs

3.1 Space Curve Construction

How to find the appropriate function:

- Interpolation vs approximation
- Complexity (computational complexity)
  - *piecewise, polynomial*
- Continuity
  - *C1, C2*
- Global vs local control

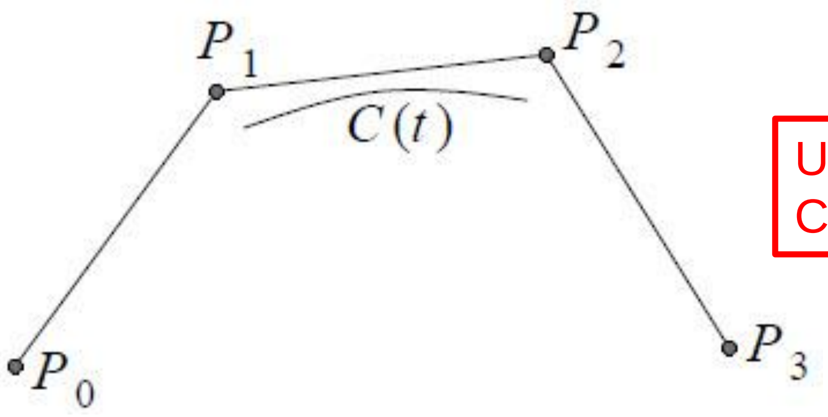
Independent of shape complexity

Affect local shape only

3.1 Space Curve Construction

Typical space curve generation technique :

- uniform cubic B-spline curve interpolation



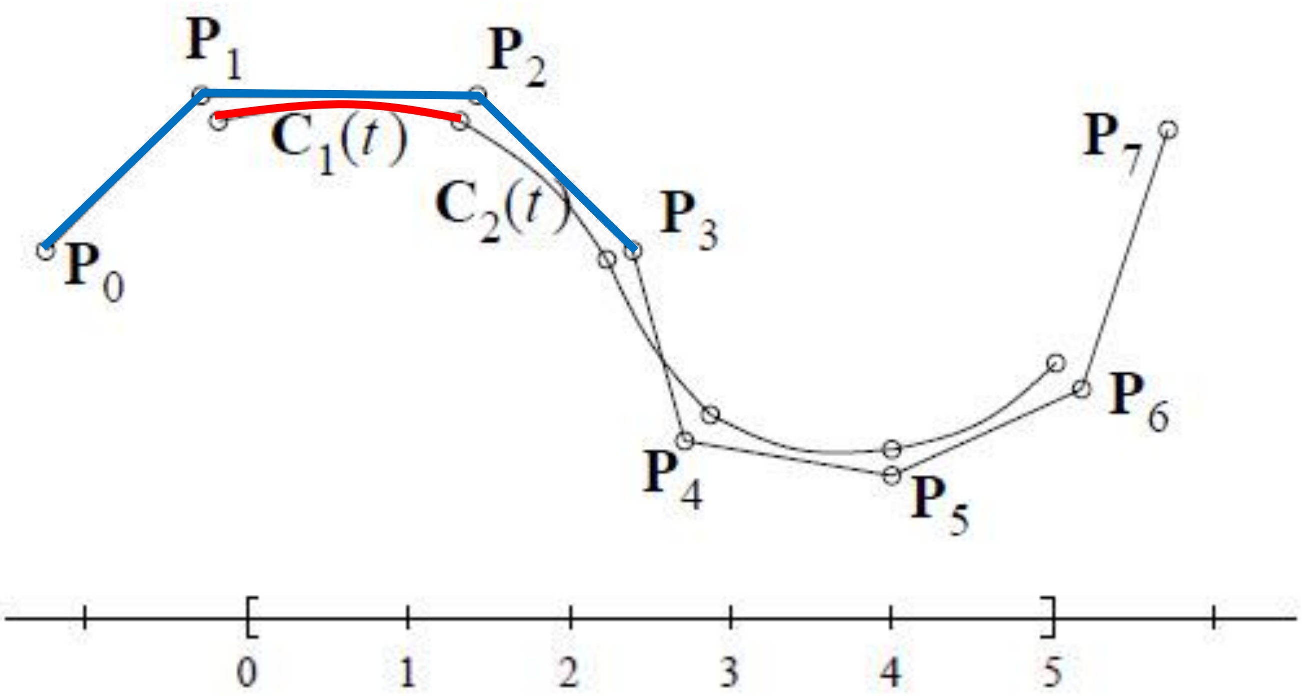
Uniform cubic B-spline Curve segment

$$C(t) = \frac{(1-t)^3}{6} P_0 + \frac{4-6t^2+3t^3}{6} P_1 + \frac{1+3t+3t^2-3t^3}{6} P_2 + \frac{t^3}{6} P_3$$

$$0 \le t \le 1$$

3.1 Space Curve Construction

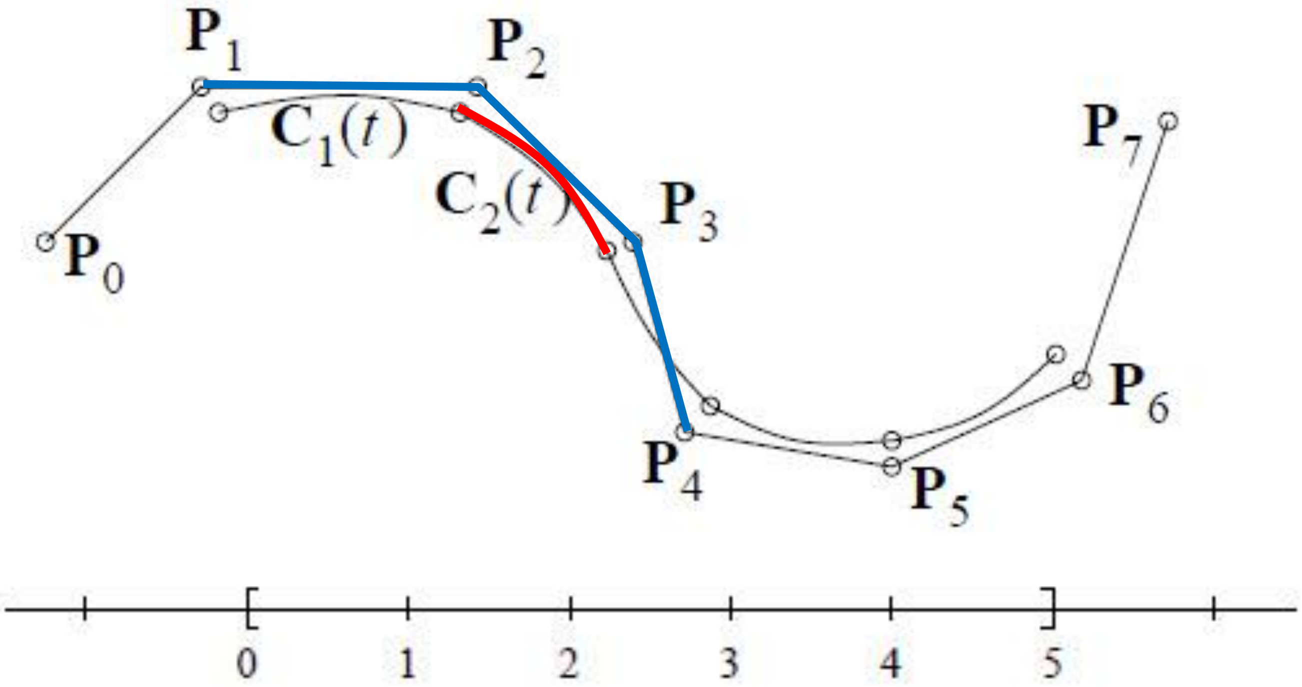
A uniform cubic B-spline curve:



$P_0, P_1, \ldots, P_7$ are given

3.1 Space Curve Construction

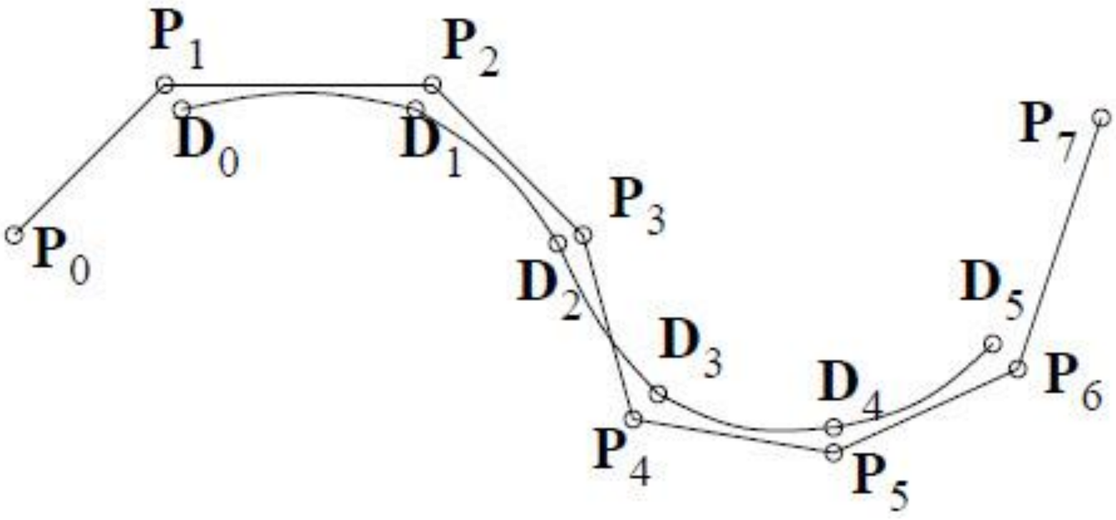
A uniform cubic B-spline curve:



$P_0, P_1, \ldots, P_7$ are given

3.1 Space Curve Construction

Reverse process:



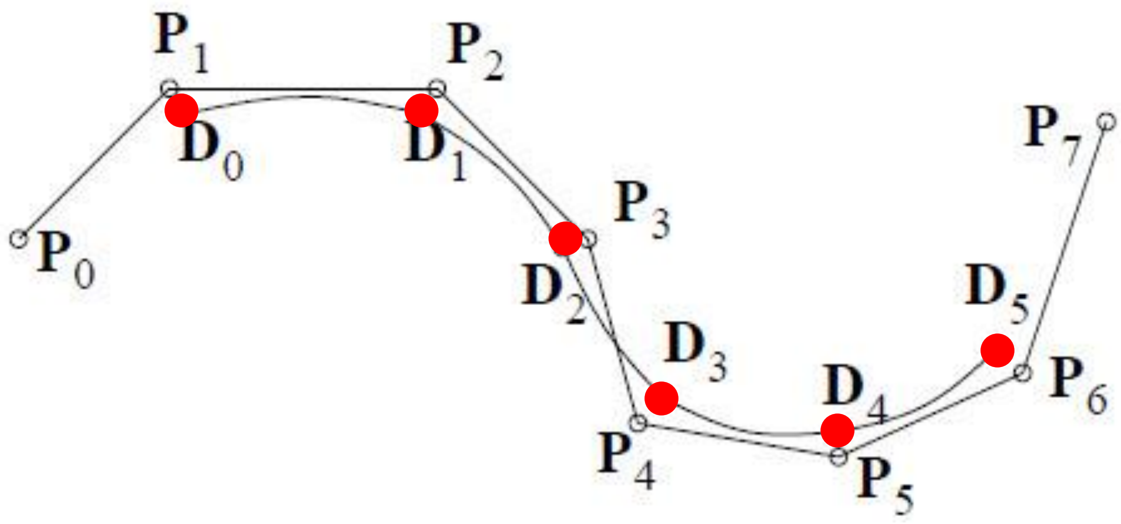
Given:

$D_0, D_1, \ldots, D_5$

Find $P_0, P_1, \ldots, P_7$ so that the corresponding uniform cubic B-spline curve would interpolate $D_0, D_1, \ldots, D_5$.

3.1 Space Curve Construction

Reverse process:



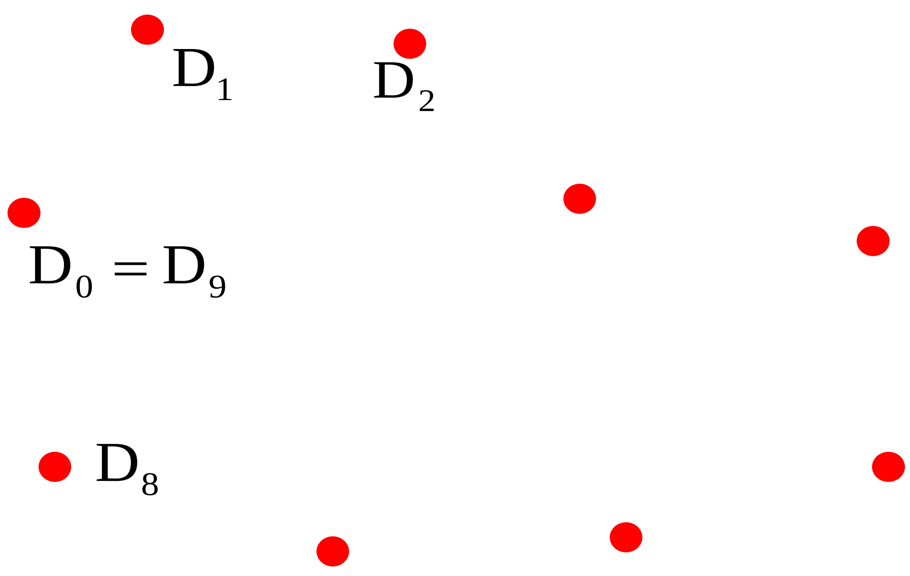
Given :

$D_0, D_1, \ldots, D_5$

Find $P_0, P_1, \ldots, P_7$ so that the corresponding uniform cubic B-spline curve would interpolate $D_0, D_1, \ldots, D_5$.

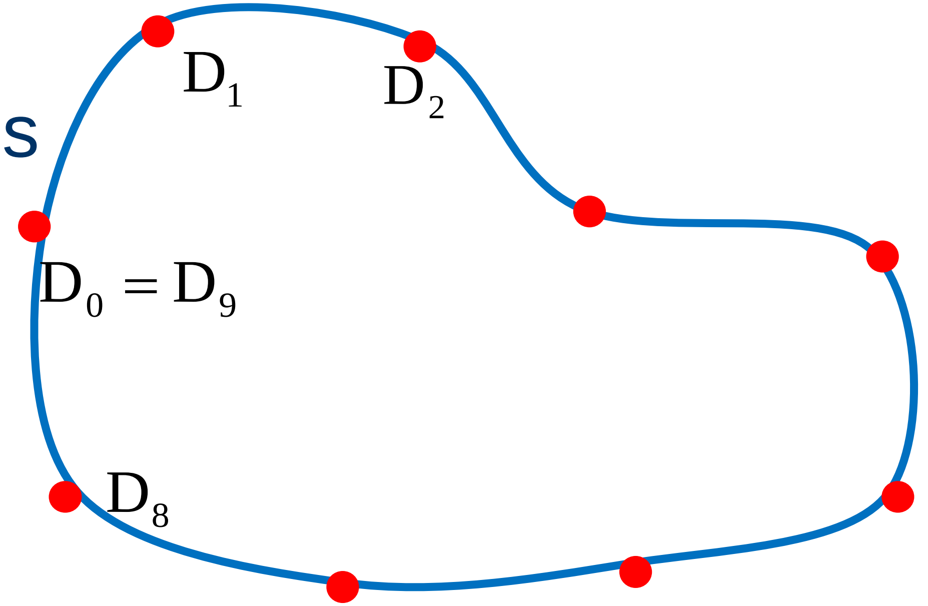
Closed Curve Fitting using Uniform Cubic B-spline Curves

Given data points $D_i = (x_i, y_i),\ i = 0, 1, \ldots, n, (n \geq 3),\ D_0 = D_n,$



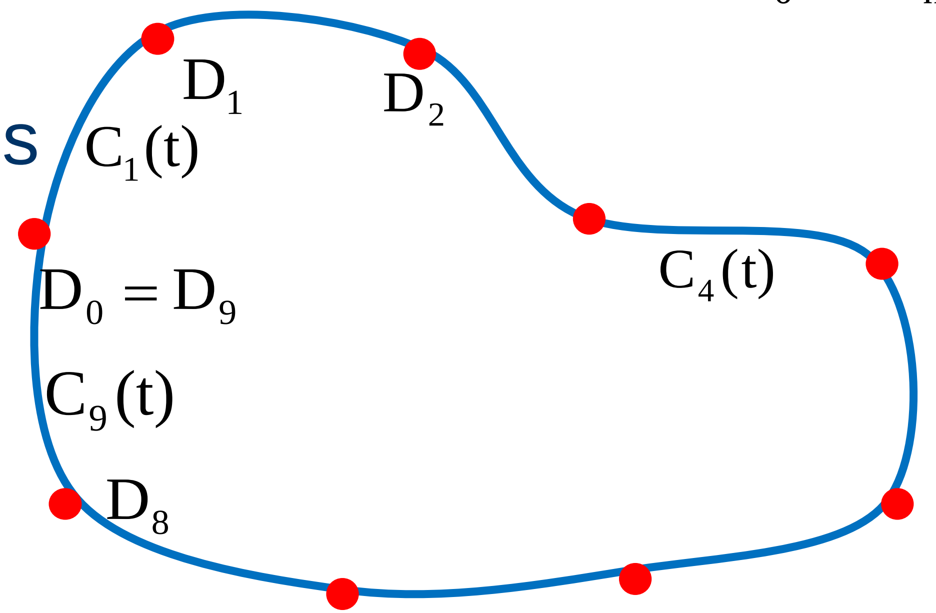
Closed Curve Fitting using Uniform Cubic B-spline Curves

Given data points $D_i = (x_i, y_i),\ i = 0, 1, \ldots, n, (n \geq 3),\ D_0 = D_n,$ how can a cubic B-spline curve S(u) that interpolates these points be constructed?



Closed Curve Fitting using Uniform Cubic B-spline Curves

Given data points $\mathrm{D_i} = (x_i, y_i),\ i = 0, 1, ..., n, (n \geq 3),\ \mathrm{D_0} = \mathrm{D_n},$ how can a cubic B-spline curve S(u) that interpolates these points be constructed?



The cubic B-spline curve has *n segments* $\mathrm{C_1}(t), C_2(t), ..., C_n(t)$ with $\mathrm{D_{i\text{-}1}}$ and $\mathrm{D_i}$ being the start and end points of $\mathrm{C_i}(t)$

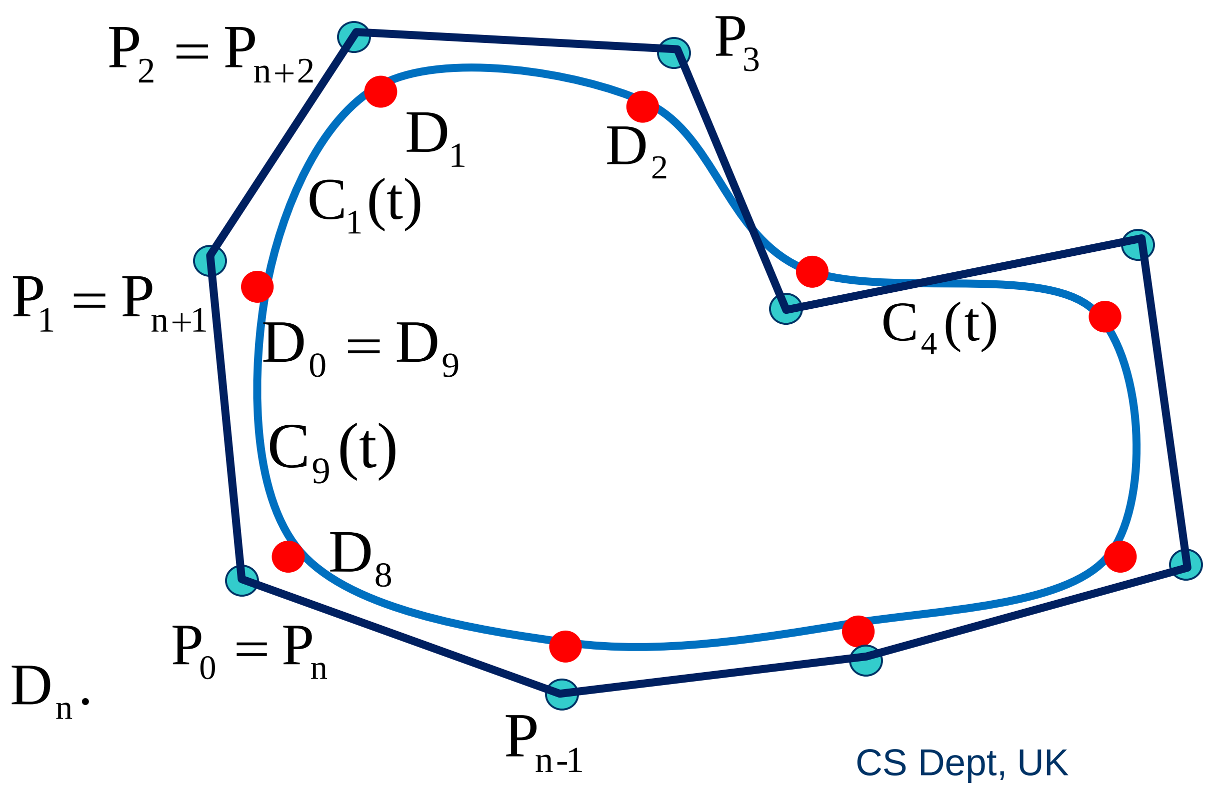
Closed Curve Fitting using Uniform Cubic B-spline Curves

The curve must have n+3 control points $P_i, i = 0, 1, ..., n+2,$ with

$$P_0 = P_n \,,$$

$$P_1 = P_{n+1} \,,$$

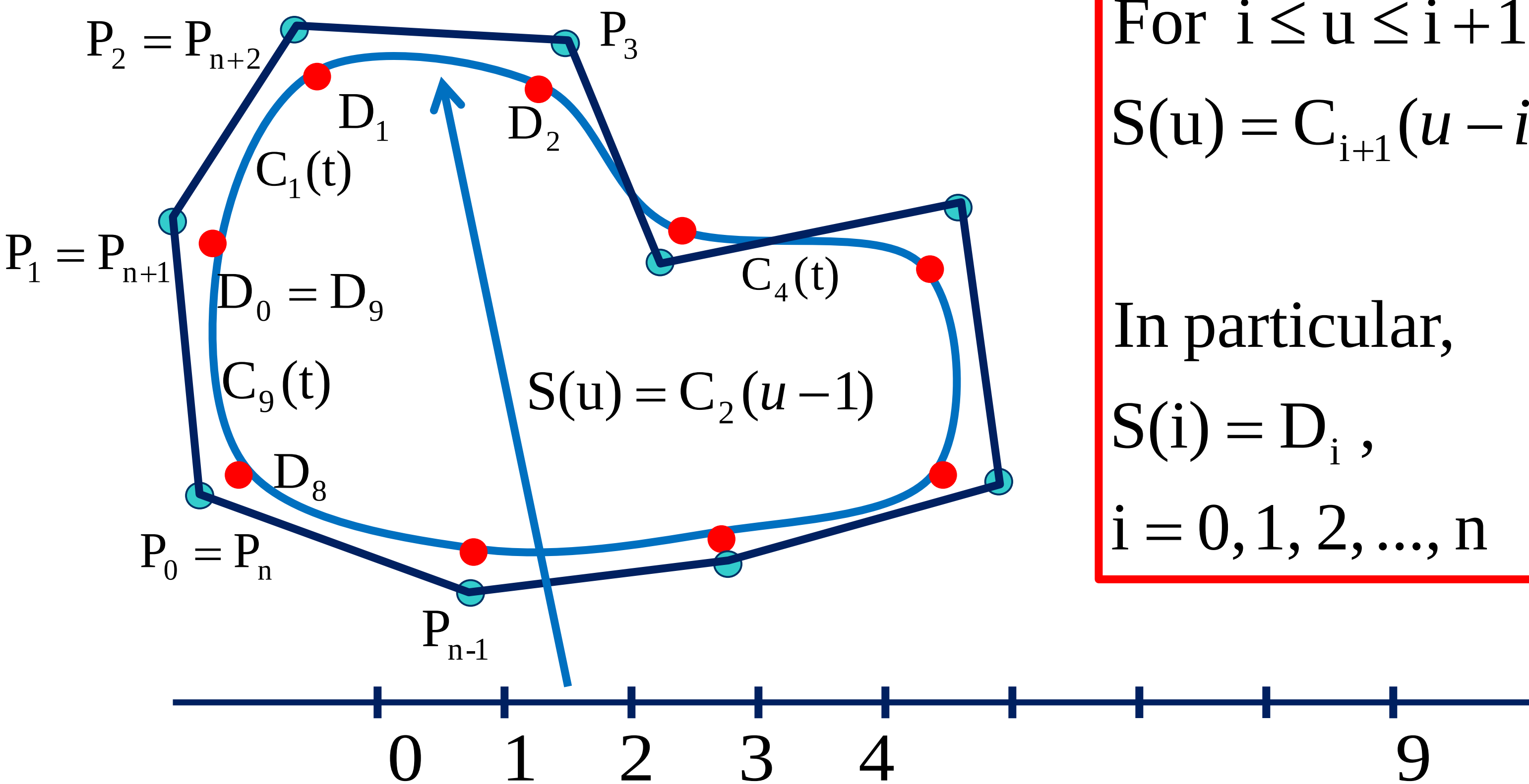
$$P_2 = P_{n+2}$$



to ensure C2-continuity at $D_0 = D_n$.

Closed Curve Fitting using Uniform Cubic B-spline Curves

Parameter space : [0, n]



For $i \le u \le i+1$,

$S(u) = C_{i+1}(u-i)$

In particular,

$S(i) = D_i$,

$i = 0, 1, 2, \ldots, n$

Closed Curve Fitting using Uniform Cubic B-spline Curves

We have

$$S(0) = D_0 \qquad P_0 + 4P_1 + P_2 = 6D_0$$

$$S(1) = D_1 \qquad P_1 + 4P_2 + P_3 = 6D_1$$

$$\vdots$$

$$S(n-1) = D_{n-1} \qquad P_{n-1} + 4P_n + P_{n+1} = 6D_{n-1}$$

$$S(n) = D_n \qquad P_n + 4P_{n+1} + P_{n+2} = 6D_n$$

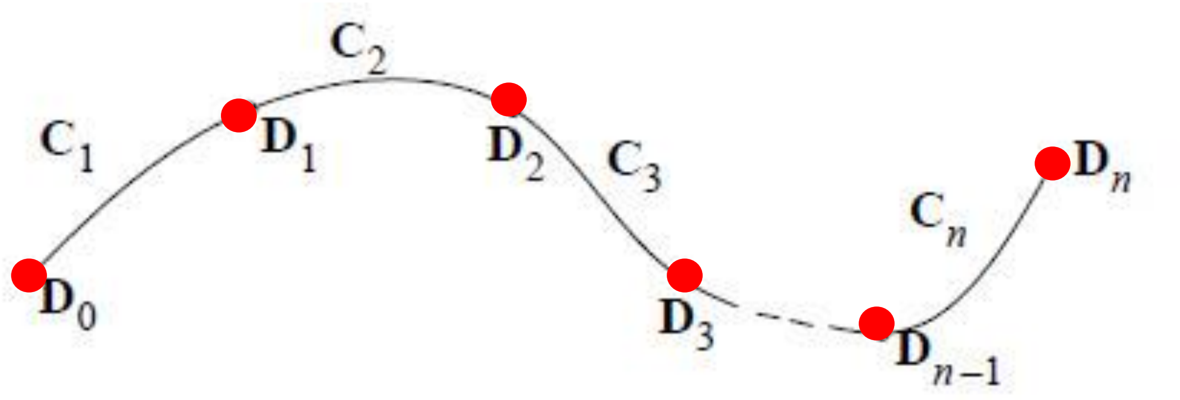
Closed Curve Fitting using Uniform Cubic B-spline Curves

In matrix form

$$
\begin{bmatrix}
4 & 1 & & & & & & 1 \\
1 & 4 & 1 & & & & & \\
& 1 & 4 & 1 & & & & \\
& & & \cdot & & & & \\
& & & \cdot & & & & \\
& & & & 1 & 4 & 1 & \\
& & & & & 1 & 4 & 1 \\
1 & & & & & & 1 & 4
\end{bmatrix}
\begin{bmatrix}
P_1 \\ P_2 \\ P_3 \\ \cdot \\ \cdot \\ P_{n-2} \\ P_{n-1} \\ P_n
\end{bmatrix}
=
\begin{bmatrix}
6D_0 \\ 6D_1 \\ 6D_2 \\ \cdot \\ \cdot \\ 6D_{n-3} \\ 6D_{n-2} \\ 6D_{n-1}
\end{bmatrix}
$$

Open Curve Fitting using Uniform Cubic B-spline Curves

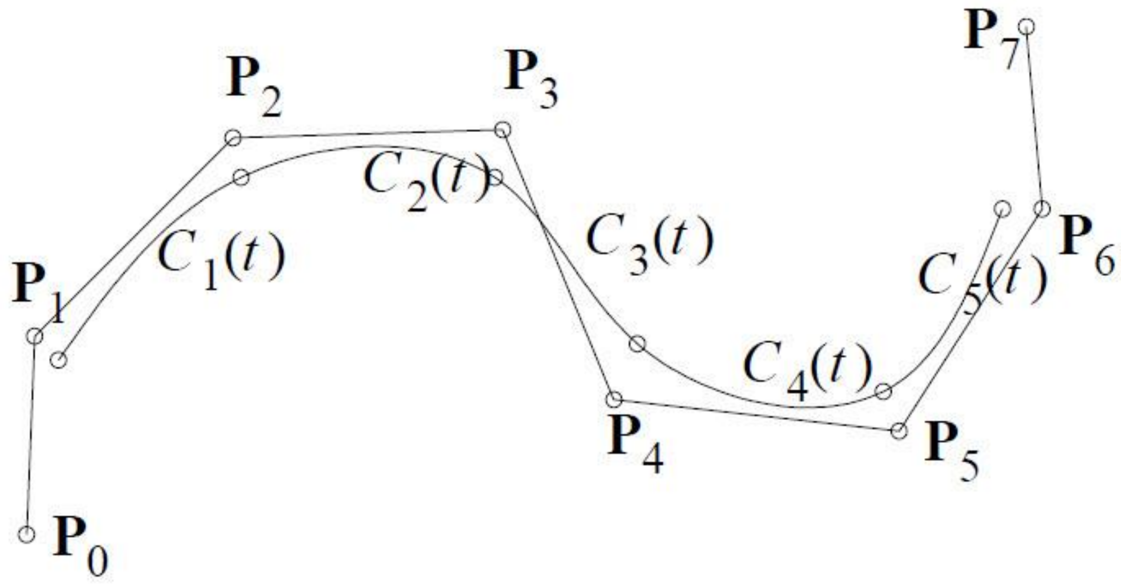
Given data points $\mathrm{D_i} = (x_i, y_i),\ i = 0, 1, ..., n, (n \geq 3)$, how can an open cubic B-spline curve S(u) that interpolates these points be built?



The cubic B-spline curve has *n segments* $\mathrm{C_1}(t), C_2(t), ..., C_n(t)$ with $\mathrm{D_{i\text{-}1}}$ and $\mathrm{D_i}$ being the start and end points of $\mathrm{C_i}(t)$

Open Curve Fitting using Uniform Cubic B-spline Curves

To get the curve constructed, how many control points are needed? Consider the following case:



n+3 control points: $P_i, i = 0, 1, 2, ..., n+2$

Open Curve Fitting using Uniform Cubic B-spline Curves

Parameter space: [0, n]

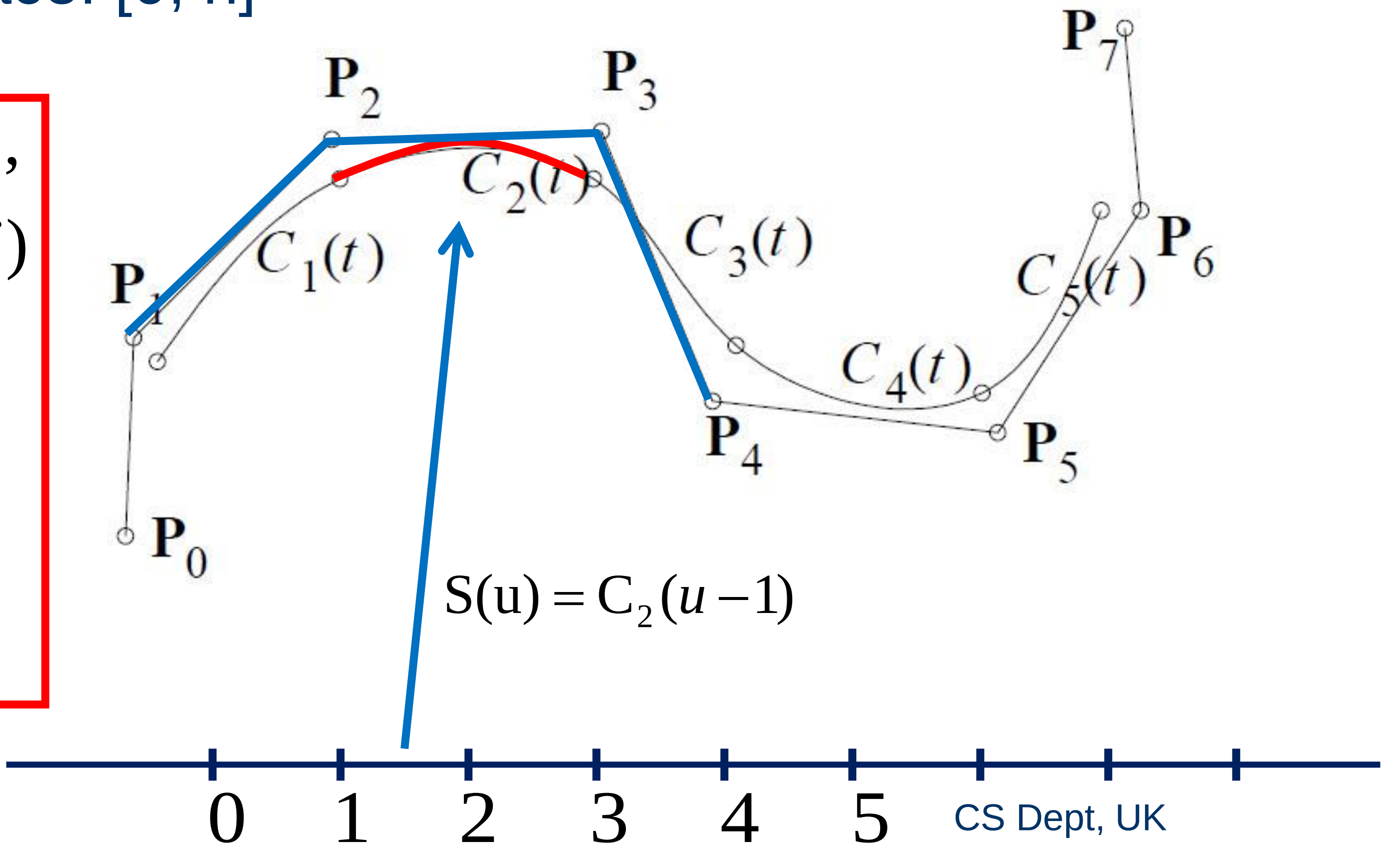
For $i \le u \le i+1$,

$S(u) = C_{i+1}(u-i)$

In particular,

$S(i) = D_i$,

$i = 0, 1, 2, \ldots, n$



Open Curve Fitting using Uniform Cubic B-spline Curves

We have

$$S(0) = D_0 \qquad P_0 + 4P_1 + P_2 = 6D_0$$

$$S(1) = D_1 \qquad P_1 + 4P_2 + P_3 = 6D_1$$

$$\vdots$$

$$S(n-1) = D_{n-1} \qquad P_{n-1} + 4P_n + P_{n+1} = 6D_{n-1}$$

$$S(n) = D_n \qquad P_n + 4P_{n+1} + P_{n+2} = 6D_n$$

Open Curve Fitting using Uniform Cubic B-spline Curves

Need two more conditions. We set $S''(0)=0, \ \ S''(n)=0$

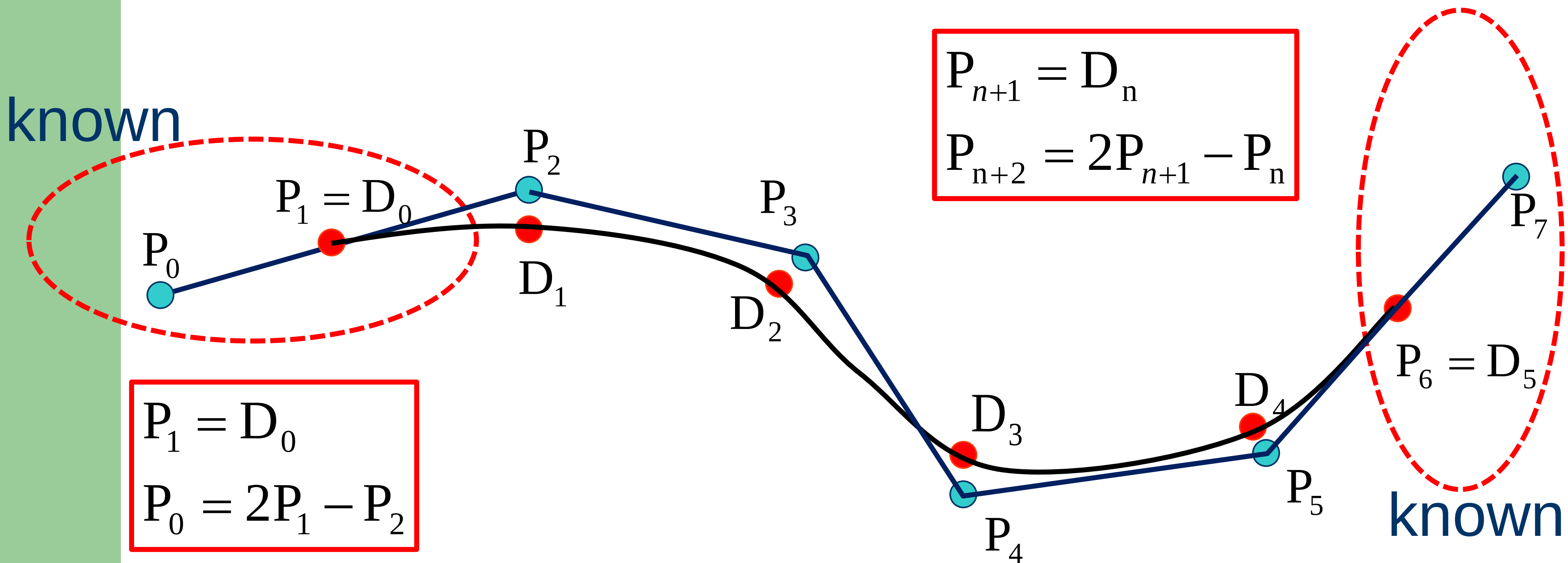
we get

$$P_0 - 2P_1 + P_2 = 0 \quad \text{and} \quad P_1 = D_0$$

$$P_n - 2P_{n+1} + P_{n+2} = 0 \quad \text{and} \quad P_{n+1} = D_n$$

Open Curve Fitting using Uniform Cubic B-spline Curves

Relationship between data points and control points:



(n=5)

Open Curve Fitting using Uniform Cubic B-spline Curves

We have

$$4P_2 + P_3 = 6D_1 - D_0$$

$$P_2 + 4P_3 + P_4 = 6D_2$$

$$\vdots$$

$$P_{n-2} + 4P_{n-1} + P_n = 6D_{n-2}$$

$$P_{n-1} + 4P_n = 6D_{n-1} - D_n$$

The matrix form of this system is:

$$
\begin{bmatrix}
4 & 1 & & & & & & \\
1 & 4 & 1 & & & & & \\
 & 1 & 4 & 1 & & & & \\
 & & & \cdot & & & & \\
 & & & \cdot & & & & \\
 & & & 1 & 4 & 1 & & \\
 & & & & 1 & 4 & 1 \\
 & & & & & 1 & 4
\end{bmatrix}
\begin{bmatrix}
P_2 \\ P_3 \\ P_4 \\ \cdot \\ \cdot \\ P_{n-2} \\ P_{n-1} \\ P_n
\end{bmatrix}
=
\begin{bmatrix}
6D_1 - D_0 \\ 6D_2 \\ 6D_3 \\ \cdot \\ \cdot \\ 6D_{n-3} \\ 6D_{n-2} \\ 6D_{n-1} - D_n
\end{bmatrix}
$$

This system of equations can be solved using Gaussian elimination without pivoting.

End of Interpolation I