Parametric Curves

Michael Kazhdan

(601.457/657)

HB 10.6 -- 10.9, 10.13
FvDFH 11.2
Overview

• What is a Spline?

• Specific Examples:
  ○ Hermite Splines
  ○ Cardinal Splines
  ○ Uniform Cubic B-Splines

• Comparing Cardinal and Uniform Cubic B-Splines
What is a Spline in CG?

A spline is a *piecewise polynomial function* whose derivatives satisfy some *continuity constraints* across curve boundaries.

\[ P_k(u) \quad u \in [0,1) \]

\[ P_1(u) \quad u \in [0,1) \]

\[ P_2(u) \quad u \in [0,1) \]

\[ P_3(x) \quad u \in [0,1) \]

\[ P_k(u) = \sum_{j=0}^{n} a_{kj} \cdot u^j \text{ with } a_{kj} \in \mathbb{R}^d \]
What is a Spline in CG?

A spline is a *piecewise polynomial function* whose derivatives satisfy some *continuity constraints* across curve boundaries.

\[ P_k(u) = \sum_{j=0}^{n} a_{kj} \cdot u^j \] with \( a_{kj} \in \mathbb{R}^d \)
Overview

• What is a Spline?

• Specific Examples:
  ◦ Hermite Splines
  ◦ Cardinal Splines
  ◦ Uniform Cubic B-Splines

• Comparing Cardinal and Uniform Cubic B-Splines
Specific Example: Hermite Splines

- **Interpolating piecewise** cubic polynomial, each specified by:
  - Start/end positions
  - Start/end tangents

- Iteratively construct the curve between adjacent end points that interpolate positions and tangents.
Specific Example: Hermite Splines

- **Interpolating** piecewise *cubic* polynomial, each specified by:
  - Start/end positions
  - Start/end tangents

- Iteratively construct the curve between adjacent end points that interpolate positions and tangents.
Specific Example: Hermite Splines

- **Interpolating** piecewise *cubic* polynomial, each specified by:
  - Start/end positions
  - Start/end tangents

- Iteratively construct the curve between adjacent end points that interpolate positions and tangents.

\[ p_0 \rightarrow p_1 \rightarrow p_2 \rightarrow \ldots \rightarrow p_4 \rightarrow p_5 \]
Specific Example: Hermite Splines

- **Interpolating** piecewise *cubic* polynomial, each specified by:
  - Start/end positions
  - Start/end tangents

- Iteratively construct the curve between adjacent end points that interpolate positions and tangents.
Specific Example: Hermite Splines

- **Interpolating** piecewise *cubic* polynomial, each specified by:
  - Start/end positions
  - Start/end tangents

- Iteratively construct the curve between adjacent end points that interpolate positions and tangents.
Specific Example: Hermite Splines

- **Interpolating** piecewise *cubic* polynomial, each specified by:
  - Start/end positions
  - Start/end tangents

- Iteratively construct the curve between adjacent end points that interpolate positions and tangents.
Specific Example: Hermite Splines

- **Interpolating** piecewise *cubic* polynomial, each specified by:
  - Start/end positions
  - Start/end tangents

- Iteratively construct the curve between adjacent end points that interpolate positions and tangents.

Because end-points of adjacent curves have the same position and derivatives, the Hermite spline is $C^1$ by construction.
Specific Example: Hermite Splines

Given the polynomial:
\[ P_k(u) = a \cdot u^3 + b \cdot u^2 + c \cdot u + d \]
we can write its derivative as:
\[ P_k'(u) = 3 \cdot a \cdot u^2 + 2 \cdot b \cdot u + c \]

Using the matrix representations:

\[
\begin{align*}
P_k(u) &= (u^3, u^2, u, 1) \cdot \begin{pmatrix} a \\ b \\ c \\ d \end{pmatrix} \\
P_k'(u) &= (3 \cdot u^2, 2 \cdot u, 1, 0) \cdot \begin{pmatrix} a \\ b \\ c \\ d \end{pmatrix}
\end{align*}
\]
Specific Example: Hermite Splines

\[ P_k(u) = (u^3 \quad u^2 \quad u \quad 1) \cdot \begin{pmatrix} a \\ b \\ c \\ d \end{pmatrix} \quad P'_k(u) = (3 \cdot u^2 \quad 2 \cdot u \quad 1 \quad 0) \cdot \begin{pmatrix} a \\ b \\ c \\ d \end{pmatrix} \]

The values/derivatives at the end-points are:

\[ \mathbf{p}_k = P_k(0) = (0 \quad 0 \quad 0 \quad 1) \cdot \begin{pmatrix} a \\ b \\ c \\ d \end{pmatrix} \quad \mathbf{t}_k = P'_k(0) = (0 \quad 0 \quad 1 \quad 0) \cdot \begin{pmatrix} a \\ b \\ c \\ d \end{pmatrix} \]

\[ \mathbf{p}_{k+1} = P_k(1) = (1 \quad 1 \quad 1 \quad 1) \cdot \begin{pmatrix} a \\ b \\ c \\ d \end{pmatrix} \quad \mathbf{t}_{k+1} = P'_k(1) = (3 \quad 2 \quad 1 \quad 0) \cdot \begin{pmatrix} a \\ b \\ c \\ d \end{pmatrix} \]
Specific Example: Hermite Splines

\[
\mathbf{p}_k = \mathbf{p}_k(0) = (0 \ 0 \ 0 \ 1) \cdot \begin{pmatrix} a \\ b \\ c \\ d \end{pmatrix} \quad \mathbf{\tilde{t}}_k = \mathbf{p}'_k(0) = (0 \ 0 \ 1 \ 0) \cdot \begin{pmatrix} a \\ b \\ c \\ d \end{pmatrix}
\]

\[
\mathbf{p}_{k+1} = \mathbf{p}_k(1) = (1 \ 1 \ 1 \ 1) \cdot \begin{pmatrix} a \\ b \\ c \\ d \end{pmatrix} \quad \mathbf{\tilde{t}}_{k+1} = \mathbf{p}'_k(1) = (3 \ 2 \ 1 \ 0) \cdot \begin{pmatrix} a \\ b \\ c \\ d \end{pmatrix}
\]

We can combine the equations into a single matrix expression:

\[
\begin{pmatrix} \mathbf{p}_k \\ \mathbf{p}_{k+1} \\ \mathbf{\tilde{t}}_k \\ \mathbf{\tilde{t}}_{k+1} \end{pmatrix} = \begin{pmatrix} 0 & 0 & 0 & 1 \\ 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 0 \\ 3 & 2 & 1 & 0 \end{pmatrix} \begin{pmatrix} a \\ b \\ c \\ d \end{pmatrix}
\]
Specific Example: Hermite Splines

\[
\begin{pmatrix}
    p_k \\
    p_{k+1} \\
    \hat{t}_k \\
    \hat{t}_{k+1}
\end{pmatrix}
= 
\begin{pmatrix}
    0 & 0 & 0 & 1 \\
    1 & 1 & 1 & 1 \\
    0 & 0 & 1 & 0 \\
    3 & 2 & 1 & 0
\end{pmatrix}
\begin{pmatrix}
    a \\
    b \\
    c \\
    d
\end{pmatrix}
\]

Inverting, we get:

\[
\begin{pmatrix}
    a \\
    b \\
    c \\
    d
\end{pmatrix}
= 
\begin{pmatrix}
    0 & 0 & 0 & 1 \\
    1 & 1 & 1 & 1 \\
    0 & 0 & 1 & 0 \\
    3 & 2 & 1 & 0
\end{pmatrix}^{-1}
\begin{pmatrix}
    p_k \\
    p_{k+1} \\
    \hat{t}_k \\
    \hat{t}_{k+1}
\end{pmatrix}
= 
\begin{pmatrix}
    2 & -2 & 1 & 1 \\
    -3 & 3 & -2 & -1 \\
    0 & 0 & 1 & 0 \\
    1 & 0 & 0 & 0
\end{pmatrix}
\begin{pmatrix}
    p_k \\
    p_{k+1} \\
    \hat{t}_k \\
    \hat{t}_{k+1}
\end{pmatrix}
\]
Specific Example: Hermite Splines

\[
\begin{pmatrix}
    a \\
    b \\
    c \\
    d
\end{pmatrix} =
\begin{pmatrix}
    2 & -2 & 1 & 1 \\
    -3 & 3 & -2 & -1 \\
    0 & 0 & 1 & 0 \\
    1 & 0 & 0 & 0
\end{pmatrix}
\begin{pmatrix}
    p_k \\
    p_{k+1} \\
    \vec{t}_k \\
    \vec{t}_{k+1}
\end{pmatrix}
\]

Using the fact that:

\[
P_k(u) = (u^3 \quad u^2 \quad u \quad 1) \cdot
\begin{pmatrix}
    a \\
    b \\
    c \\
    d
\end{pmatrix}
\]

We get:

\[
P_k(u) = (u^3 \quad u^2 \quad u \quad 1)
\begin{pmatrix}
    2 & -2 & 1 & 1 \\
    -3 & 3 & -2 & -1 \\
    0 & 0 & 1 & 0 \\
    1 & 0 & 0 & 0
\end{pmatrix}
\begin{pmatrix}
    p_k \\
    p_{k+1} \\
    \vec{t}_k \\
    \vec{t}_{k+1}
\end{pmatrix}
\]
Specific Example: Hermite Splines

Setting:

- $H_0(u) = 2u^3 - 3u^2 + 1$
- $H_1(u) = -2u^3 + 3u^2$
- $H_2(u) = u^3 - 2u^2 + u$
- $H_3(u) = u^3 - u^2$

Blending Functions

$$P_k(u) = H_0(u) \cdot p_k + H_1(u) \cdot p_{k+1} + H_2(u) \cdot \hat{t}_k + H_3(u) \cdot \hat{t}_{k+1}$$
Specific Example: Hermite Splines

- **Interpolating** piecewise *cubic* polynomial, each specified by:
  - Start/end positions
  - Start/end tangents

- Iteratively construct the curve between adjacent end points that interpolate positions and tangents.

Given the control points, how do we define the value of the tangents/derivatives?
Overview

• What is a Spline?

• Specific Examples:
  ○ Hermite Splines
  ○ Cardinal Splines
  ○ Uniform Cubic B-Splines

• Comparing Cardinal and Uniform Cubic B-Splines
Specific Example: Cardinal Splines

- Interpolating piecewise cubic polynomial, each specified by four control points.
- Turn into a Hermite problem by iteratively constructing the curve between middle two points using adjacent points to define tangents.
Specific Example: Cardinal Splines

- Interpolating piecewise cubic polynomial, each specified by four control points.
- Turn into a Hermite problem by iteratively constructing the curve between middle two points using adjacent points to define tangents.
Specific Example: Cardinal Splines

• **Interpolating** piecewise *cubic* polynomial, each specified by four control points.

• Turn into a Hermite problem by iteratively constructing the curve between middle two points *using adjacent points to define tangents*. 
Specific Example: Cardinal Splines

• **Interpolating** piecewise *cubic* polynomial, each specified by four control points.

• Turn into a Hermite problem by iteratively constructing the curve between middle two points *using adjacent points to define tangents*
Specific Example: Cardinal Splines

- **Interpolating** piecewise *cubic* polynomial, each specified by four control points.

- Turn into a Hermite problem by iteratively constructing the curve between middle two points *using adjacent points to define tangents*.
Specific Example: Cardinal Splines

- **Interpolating** piecewise cubic polynomial, each specified by four control points.

- Turn into a Hermite problem by iteratively constructing the curve between middle two points using adjacent points to define tangents.

![Diagram of Cardinal Splines](image-url)

\[ \mathbf{p}_0, \mathbf{p}_1, \mathbf{p}_2, \mathbf{p}_3, \mathbf{p}_4, \mathbf{p}_5, \mathbf{p}_6, \mathbf{p}_7 \]
Specific Example: Cardinal Splines

- **Interpolating** piecewise cubic polynomial, each specified by four control points.

- Turn into a Hermite problem by iteratively constructing the curve between middle two points using adjacent points to define tangents.
Specific Example: Cardinal Splines

• **Interpolating** piecewise *cubic* polynomial, each specified by four control points.

• Turn into a Hermite problem by iteratively constructing the curve between middle two points *using adjacent points to define tangents*.
Specific Example: Cardinal Splines

- Interpolating piecewise cubic polynomial, each specified by four control points.
- Turn into a Hermite problem by iteratively constructing the curve between middle two points using adjacent points to define tangents.
Specific Example: Cardinal Splines

- **Interpolating** piecewise *cubic* polynomial, each specified by four control points.

- Turn into a Hermite problem by iteratively constructing the curve between middle two points *using adjacent points to define tangents*.
Specific Example: Cardinal Splines

- **Interpolating** piecewise *cubic* polynomial, each specified by four control points.

- Turn into a Hermite problem by iteratively constructing the curve between middle two points *using adjacent points to define tangents*. 
Specific Example: Cardinal Splines

- **Interpolating** piecewise *cubic* polynomial, each specified by four control points.
- Turn into a Hermite problem by iteratively constructing the curve between middle two points *using adjacent points to define tangents*
Specific Example: Cardinal Splines

- Interpolating piecewise cubic polynomial, each specified by four control points.
- Turn into a Hermite problem by iteratively constructing the curve between middle two points using adjacent points to define tangents.

![Cardinal Spline Example Diagram](image-url)
Specific Example: Cardinal Splines

• Interpolating piecewise cubic polynomial, each specified by four control points.

• Turn into a Hermite problem by iteratively constructing the curve between middle two points using adjacent points to define tangents

Because the end-points of adjacent curves share the same position and derivatives, the Cardinal spline has $C^1$ continuity.
Specific Example: Cardinal Splines

Using Hermite splines, we have:

\[ P_k(u) = (u^3 \quad u^2 \quad u \quad 1) \begin{pmatrix} 2 & -2 & 1 & 1 \\ -3 & 3 & -2 & -1 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} \vec{p}_k \\ \vec{p}_{k+1} \\ \vec{t}_k \\ \vec{t}_{k+1} \end{pmatrix} \]

\[ \vec{t}_k = s(\vec{p}_{k+1} - \vec{p}_{k-1}) \]

\[ \vec{t}_{k+1} = s(\vec{p}_{k+2} - \vec{p}_k) \]
Specific Example: Cardinal Splines

We can express the boundary constraints as:

\[
\begin{pmatrix}
  \mathbf{p}_k \\
  \mathbf{p}_{k+1} \\
  \hat{t}_k \\
  \hat{t}_{k+1}
\end{pmatrix}
= 
\begin{pmatrix}
  \mathbf{p}_k \\
  \mathbf{p}_{k+1} \\
  s(\mathbf{p}_{k+1} - \mathbf{p}_{k-1}) \\
  s(\mathbf{p}_{k+2} - \mathbf{p}_k)
\end{pmatrix}
= 
\begin{pmatrix}
  0 & 1 & 0 & 0 \\
  0 & 0 & 1 & 0 \\
  -s & 0 & s & 0 \\
  0 & -s & 0 & s
\end{pmatrix}
\begin{pmatrix}
  \mathbf{p}_{k-1} \\
  \mathbf{p}_k \\
  \mathbf{p}_{k+1} \\
  \mathbf{p}_{k+2}
\end{pmatrix}
\]

So using the approach of Hermite spline, we get:

\[
\mathbf{P}_k(u) = (u^3 \quad u^2 \quad u \quad 1)
\begin{pmatrix}
  2 & -2 & 1 & 1 \\
  -3 & 3 & -2 & -1 \\
  0 & 0 & 1 & 0 \\
  1 & 0 & 0 & 0
\end{pmatrix}
\begin{pmatrix}
  0 & 1 & 0 & 0 \\
  0 & 0 & 1 & 0 \\
  -s & 0 & s & 0 \\
  0 & -s & 0 & s
\end{pmatrix}
\begin{pmatrix}
  \mathbf{p}_{k-1} \\
  \mathbf{p}_k \\
  \mathbf{p}_{k+1} \\
  \mathbf{p}_{k+2}
\end{pmatrix}
\]

\[
\mathbf{M}_{\text{Hermite}}
\]
Specific Example: Cardinal Splines

\[
P_k(u) = (u^3 \quad u^2 \quad u \quad 1) \begin{pmatrix} 2 & -2 & 1 & 1 \\ -3 & 3 & -2 & -1 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ -s & 0 & s & 0 \\ 0 & -s & 0 & s \end{pmatrix} \begin{pmatrix} p_{k-1} \\ p_k \\ p_{k+1} \\ p_{k+2} \end{pmatrix}
\]

Multiplying, we get the Cardinal matrix representation:

\[
P_k(u) = (u^3 \quad u^2 \quad u \quad 1) \begin{pmatrix} -s & 2 - s & s - 2 & s \\ 2s & s - 3 & 3 - 2s & -s \\ -s & 0 & s & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix} \begin{pmatrix} p_{k-1} \\ p_k \\ p_{k+1} \\ p_{k+2} \end{pmatrix}
\]

\(M_{\text{Cardinal}}\)
Specific Example: Cardinal Splines

Setting:

- \( C_0(u) = -su^3 + 2su^2 - su \)
- \( C_1(u) = (2 - s)u^3 + (s - 3)u^2 + 1 \)
- \( C_2(u) = (s - 2)u^3 + (3 - 2s)u^2 + su \)
- \( C_3(u) = su^3 - su^2 \)

For \( s = 1/2 \):

\[
P_k(u) = C_0(u) \cdot p_{k-1} + C_1(u) \cdot p_k + C_2(u) \cdot p_{k+1} + C_3(u) \cdot p_{k+2}
\]
Specific Example: Cardinal Splines

- Interpolating piecewise cubic polynomial, each specified by four control points.

- Iteratively construct the curve between middle two points using adjacent points to define tangents.

At the first and last end-points, you can:
- Not draw the final segments
- Double up end points
- Loop the spline around
Overview

• What is a Spline?

• Specific Examples:
  ○ Hermite Splines
  ○ Cardinal Splines
  ○ Uniform Cubic B-Splines

• Comparing Cardinal and Uniform Cubic B-Splines
Specific Example: Uniform Cubic B-Splines

• Approximating piecewise cubic polynomial, each specified by four control points.

• Iteratively construct the curve near middle two points using adjacent points to define positions and tangents.
Specific Example: Uniform Cubic B-Splines

- **Approximating** piecewise *cubic* polynomial, each specified by four control points.

- Iteratively construct the curve near middle two points using adjacent points to define positions and tangents.

\[
\mathbf{p}_0 \quad \mathbf{p}_1 \quad \mathbf{p}_2 \quad \mathbf{p}_3 \quad \mathbf{p}_4 \quad \mathbf{p}_5 \quad \mathbf{p}_6 \quad \mathbf{p}_7
\]
Specific Example: Uniform Cubic B-Splines

- Approximating piecewise cubic polynomial, each specified by four control points.
- Iteratively construct the curve near middle two points using adjacent points to define positions and tangents.
Specific Example: Uniform Cubic B-Splines

• **Approximating** piecewise *cubic* polynomial, each specified by four control points.

• Iteratively construct the curve near middle two points using adjacent points to define positions and tangents.
Specific Example: Uniform Cubic B-Splines

- **Approximating** piecewise *cubic* polynomial, each specified by four control points.
- Iteratively construct the curve near middle two points using adjacent points to define positions and tangents.

\[
\begin{align*}
&\mathbf{p}_0 \\
&\mathbf{p}_1 \\
&\mathbf{p}_2 \\
&\mathbf{p}_3 \\
&\mathbf{p}_4 \\
&\mathbf{p}_5 \\
&\mathbf{p}_6 \\
&\mathbf{p}_7
\end{align*}
\]
Specific Example: Uniform Cubic B-Splines

- **Approximating** piecewise *cubic* polynomial, each specified by four control points.
- Iteratively construct the curve near middle two points using adjacent points to define positions and tangents.
Specific Example: Uniform Cubic B-Splines

- Approximating piecewise cubic polynomial, each specified by four control points.
- Iteratively construct the curve near middle two points using adjacent points to define positions and tangents.
Specific Example: Uniform Cubic B-Splines

- Approximating piecewise cubic polynomial, each specified by four control points.
- Iteratively construct the curve near middle two points using adjacent points to define positions and tangents.
Specific Example: Uniform Cubic B-Splines

- **Approximating** piecewise *cubic* polynomial, each specified by four control points.

- Iteratively construct the curve near middle two points using adjacent points to define positions and tangents.
Specific Example: Uniform Cubic B-Splines

• **Approximating** piecewise *cubic* polynomial, each specified by four control points.

• Iteratively construct the curve near middle two points using adjacent points to define positions and tangents.
Specific Example: Uniform Cubic B-Splines

• **Approximating** piecewise *cubic* polynomial, each specified by four control points.

• Iteratively construct the curve near middle two points using adjacent points to define positions and tangents.
Specific Example: Uniform Cubic B-Splines

- **Approximating** piecewise *cubic* polynomial, each specified by four control points.
- Iteratively construct the curve near middle two points using adjacent points to define positions and tangents.
Approximating piecewise cubic polynomial, each specified by four control points.

Iteratively construct the curve near middle two points using adjacent points to define positions and tangents.

Specific Example: Uniform Cubic B-Splines

\[ p_0 \quad p_1 \quad p_2 \quad p_3 \quad p_4 \quad p_5 \quad p_6 \quad p_7 \]
Specific Example: Uniform Cubic B-Splines

- Approximating piecewise cubic polynomial, each specified by four control points.
- Iteratively construct the curve near middle two points using adjacent points to define positions and tangents.
Specific Example: Uniform Cubic B-Splines

• Approximating piecewise cubic polynomial, each specified by four control points.

• Iteratively construct the curve near middle two points using adjacent points to define positions and tangents.
Specific Example: Uniform Cubic B-Splines

- **Approximating** piecewise *cubic* polynomial, each specified by four control points.
- Iteratively construct the curve near middle two points using adjacent points to define positions and tangents.
Specific Example: Uniform Cubic B-Splines

• **Approximating** piecewise *cubic* polynomial, each specified by four control points.

• Iteratively construct the curve near middle two points using adjacent points to define positions and tangents.
Specific Example: Uniform Cubic B-Splines

• **Approximating** piecewise *cubic* polynomial, each specified by four control points.

• Iteratively construct the curve near middle two points using adjacent points to define positions and tangents.
Specific Example: Uniform Cubic B-Splines

- **Approximating** piecewise *cubic* polynomial, each specified by four control points.
- Iteratively construct the curve near middle two points using adjacent points to define positions and tangents.

```plaintext
\[ \mathbf{p}_0 \quad \mathbf{p}_1 \quad \mathbf{p}_2 \quad \mathbf{p}_3 \quad \mathbf{p}_4 \quad \mathbf{p}_5 \quad \mathbf{p}_6 \quad \mathbf{p}_7 \]
```
Specific Example: Uniform Cubic B-Splines

Using Hermite splines, we have:

\[
P_k(u) = (u^3 \quad u^2 \quad u \quad 1)
\]

\[
\begin{pmatrix}
-3 & 3 & -2 & -1 \\
0 & 0 & 1 & 0 \\
1 & 0 & 0 & 0
\end{pmatrix}
\begin{pmatrix}
\mathbf{p}'_k \\
\mathbf{p}'_{k+1} \\
\mathbf{t}_k \\
\mathbf{t}_{k+1}
\end{pmatrix}
\]

\[
\mathbf{p}'_k = \frac{(\mathbf{p}_{k-1} + 4\mathbf{p}_k + \mathbf{p}_{k+1})}{6}
\]

\[
\mathbf{p}'_{k+1} = \frac{(\mathbf{p}_k + 4\mathbf{p}_{k+1} + \mathbf{p}_{k+2})}{6}
\]

\[
\mathbf{t}_k = s(\mathbf{p}_{k+1} - \mathbf{p}_{k-1})
\]

\[
\mathbf{t}_{k+1} = s(\mathbf{p}_{k+2} - \mathbf{p}_k)
\]
Specific Example: Uniform Cubic B-Splines

We can express the boundary constraints as:

\[
\begin{pmatrix}
\mathbf{p}'_k \\
\mathbf{p}'_{k+1} \\
\mathbf{t}_k \\
\mathbf{t}_{k+1}
\end{pmatrix} = \frac{1}{6} \begin{pmatrix}
\mathbf{p}_{k-1} + 4\mathbf{p}_k + \mathbf{p}_{k+1} \\
\mathbf{p}_k + 4\mathbf{p}_{k+1} + \mathbf{p}_{k+2} \\
6s(\mathbf{p}_{k+1} - \mathbf{p}_{k-1}) \\
6s(\mathbf{p}_{k+2} - \mathbf{p}_k)
\end{pmatrix}
\]

So using the approach of Hermite spline, we get:

\[
\mathbf{p}_k(u) = (u^3 \quad u^2 \quad u \quad 1) \begin{pmatrix}
2 & -2 & 1 & 1 \\
-3 & 3 & -2 & -1 \\
0 & 0 & 1 & 0 \\
1 & 0 & 0 & 0
\end{pmatrix} \frac{1}{6} \begin{pmatrix}
1 & 4 & 1 & 0 \\
0 & 1 & 4 & 1 \\
-6s & 0 & 6s & 0 \\
1 & -6s & 0 & 6s
\end{pmatrix}\begin{pmatrix}
\mathbf{p}_{k-1} \\
\mathbf{p}_k \\
\mathbf{p}_{k+1} \\
\mathbf{p}_{k+2}
\end{pmatrix}
\]
Specific Example: Uniform Cubic B-Splines

\[ \mathbf{p}_k(u) = (u^3 \quad u^2 \quad u \quad 1) \begin{pmatrix} 2 & -2 & 1 & 1 \\ -3 & 3 & -2 & -1 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \end{pmatrix} \frac{1}{6} \begin{pmatrix} 1 & 4 & 1 & 0 \\ 0 & 1 & 4 & 1 \\ -6s & 0 & 6s & 0 \\ 1 & -6s & 0 & 6s \end{pmatrix} \begin{pmatrix} \mathbf{p}_{k-1} \\ \mathbf{p}_k \\ \mathbf{p}_{k+1} \\ \mathbf{p}_{k+2} \end{pmatrix} \]

Multiplying, we get the uniform cubic B-spline matrix representation:

\[ \mathbf{p}_k(u) = (u^3 \quad u^2 \quad u \quad 1) \frac{1}{6} \begin{pmatrix} 2 - 6s & 6 - 6s & -6 + 6s & -2 + 6s \\ -3 + 12s & -9 + 6s & 9 - 12s & 3 - 6s \\ -6s & 0 & 6s & 0 \\ 1 & 4 & 1 & 0 \end{pmatrix} \begin{pmatrix} \mathbf{p}_{k-1} \\ \mathbf{p}_k \\ \mathbf{p}_{k+1} \\ \mathbf{p}_{k+2} \end{pmatrix} \]
Specific Example: Uniform Cubic B-Splines

Setting the blending functions to:

- \( B_{0,3}(u) = (\frac{1}{3} - s)u^3 + (-\frac{1}{2} + 2s)u^2 - su + \frac{1}{6} \)
- \( B_{1,3}(u) = (1 - s)u^3 + (-\frac{3}{2} + s)u^2 + \frac{2}{3} \)
- \( B_{2,3}(u) = (-1 + s)u^3 + (\frac{3}{2} - 2s)u^2 + su + \frac{1}{6} \)
- \( B_{3,3}(u) = (-\frac{1}{3} + s)u^3 + (\frac{1}{2} - s)u^2 \)

For \( s = 1/2 \):

\[
P_k(u) = B_{0,3}(u) \cdot p_{k-1} + B_{1,3}(u) \cdot p_k + B_{2,3}(u) \cdot p_{k+1} + B_{3,3}(u) \cdot p_{k+2}
\]
Specific Example: Uniform Cubic B-Splines

- **Approximating** piecewise *cubic* polynomial, each specified by four control points.
- Iteratively construct the curve near middle two points using adjacent points to define positions and tangents.

At the first and last end-points, you can:
- Not draw the final segments
- Double up end points
- Loop the spline around
Overview

• What is a Spline?

• Specific Examples:
  ◦ Hermite Splines
  ◦ Cardinal Splines
  ◦ Uniform Cubic B-Splines

• Comparing Cardinal and Uniform Cubic B-Splines
Blending Functions

Blending functions provide a way for expressing the functions \( P_k(u) \) as a weighted sum of the four control points \( p_{k-1}, p_k, p_{k+1}, \) and \( p_{k+2} \):

\[
P_k(u) = BF_0(u) \cdot p_{k-1} + BF_1(u) \cdot p_k + BF_2(u) \cdot p_{k+1} + BF_3(u) \cdot p_{k+2}
\]
Blending Functions

Properties:

- Translation Equivariance:
  - If we translate all the control points by the same vector $\mathbf{q}$, the position of the new curve at value $u$ should be the position of the old curve at $u$, translated by $\mathbf{q}$.

  $\Rightarrow$ Given control points $\{\mathbf{p}_{k-1}, \mathbf{p}_k, \mathbf{p}_{k+1}, \mathbf{p}_{k+2}\}$ and translation vector $\mathbf{q}$:
  
  Let $\mathbf{P}_k(u)$ be the curve defined by $\{\mathbf{p}_{k-1}, \mathbf{p}_k, \mathbf{p}_{k+1}, \mathbf{p}_{k+2}\}$.
  
  Let $\mathbf{Q}_k(u)$ be the curve defined by $\{\mathbf{q} + \mathbf{p}_{k-1}, \mathbf{q} + \mathbf{p}_k, \mathbf{q} + \mathbf{p}_{k+1}, \mathbf{q} + \mathbf{p}_{k+2}\}$.
  
  We want:

  $\mathbf{Q}_k(u) = \mathbf{q} + \mathbf{P}_k(u)$

  $\Rightarrow$ Writing out $\mathbf{Q}_k(u)$, we have:

  $\mathbf{Q}_k(u) = BF_0(u)(\mathbf{q} + \mathbf{p}_{k-1}) + BF_1(u)(\mathbf{q} + \mathbf{p}_k) + BF_2(u)(\mathbf{q} + \mathbf{p}_{k+1}) + BF_3(u)(\mathbf{q} + \mathbf{p}_{k+2})$

  $= (BF_0(u) + BF_1(u) + BF_2(u) + BF_3(u))\mathbf{q} + \mathbf{P}_k(u)$

  $\Rightarrow$ To satisfy translation equivariance, we must have:

  $BF_0(u) + BF_1(u) + BF_2(u) + BF_3(u) = 1$

\[
\mathbf{P}_k(u) = BF_0(u) \cdot \mathbf{p}_{k-1} + BF_1(u) \cdot \mathbf{p}_k + BF_2(u) \cdot \mathbf{p}_{k+1} + BF_3(u) \cdot \mathbf{p}_{k+2}
\]
Comparison: Cardinal vs. Cubic B

Cardinal Splines ($s = 1/2$)

- $BF_0(u) = -\frac{1}{2}u^3 + u^2 - \frac{1}{2}u$
- $BF_1(u) = \frac{3}{2}u^3 - \frac{5}{2}u^2 + \frac{1}{2}$
- $BF_2(u) = -\frac{3}{2}u^3 + 2u^2 + \frac{1}{2}u$
- $BF_3(u) = \frac{1}{2}u^3 - \frac{1}{2}u^2$

$BF_0(u) + BF_1(u) + BF_2(u) + BF_3(u) = 1$

Cubic B-Splines ($s = 1/2$)

- $BF_0(u) = -\frac{1}{6}u^3 + \frac{1}{2}u^2 - \frac{1}{2}u + \frac{1}{6}$
- $BF_1(u) = \frac{1}{2}u^3 - u^2 + \frac{2}{3}$
- $BF_2(u) = -\frac{1}{2}u^3 + \frac{1}{2}u^2 + \frac{1}{2}u + \frac{1}{6}$
- $BF_3(u) = \frac{1}{6}u^3$

$BF_0(u) + BF_1(u) + BF_2(u) + BF_3(u) = 1$

$p_k(u) = BF_0(u) \cdot p_{k-1} + BF_1(u) \cdot p_k + BF_2(u) \cdot p_{k+1} + BF_3(u) \cdot p_{k+2}$
Blending Functions

Properties:

- **Translation Equivariance:**
  \[ BF_0(u) + BF_1(u) + BF_2(u) + BF_3(u) = 1 \text{ for all } 0 \leq u \leq 1. \]

- **Continuity:**
  
  - We need the curve \( P_{k+1}(u) \) to begin where \( P_k(u) \) ended.
  
  \[ \Rightarrow \text{ Taking the difference, we get: } 0 = P_{k+1}(0) - P_k(1) \]

  \[ \Rightarrow \text{ Expanding this out, we get: } 0 = (\ldots - BF_0(1))p_{k-1} + (BF_0(0) - BF_1(1))p_k + (BF_1(0) - BF_2(1))p_{k+1} + (BF_2(0) - BF_3(1))p_{k+2} + (BF_3(0))p_{k+3} \]

  \[ \Rightarrow \text{ For this to be true for all control points } \{p_{k-1}, p_k, p_{k+1}, p_{k+2}, p_{k+3}\}, \text{ we must have: } \]

  \[ 0 = BF_0(1) \]

  \[ BF_0(0) = BF_1(1) \]

  \[ BF_1(0) = BF_2(1) \]

  \[ BF_2(0) = BF_3(1) \]

  \[ BF_3(0) = 0 \]

\[ P_k(u) = BF_0(u) \cdot p_{k-1} + BF_1(u) \cdot p_k + BF_2(u) \cdot p_{k+1} + BF_3(u) \cdot p_{k+2} \]
Blending Functions

Properties:

More Generally, for the spline to have continuous $n$-th order derivatives, the blending functions need to satisfy:

$$0 = BF_0^{(n)}(1)$$
$$BF_0^{(n)}(0) = BF_1^{(n)}(1)$$
$$BF_1^{(n)}(0) = BF_2^{(n)}(1)$$
$$BF_2^{(n)}(0) = BF_3^{(n)}(1)$$
$$BF_3^{(n)}(0) = 0$$

⇒ For this to be true for all control points $\{p_{k-1}, p_k, p_{k+1}, p_{k+2}, p_{k+3}\}$, we must have:

$$0 = BF_0(1)$$
$$BF_0(0) = BF_1(1)$$
$$BF_1(0) = BF_2(1)$$
$$BF_2(0) = BF_3(1)$$
$$BF_3(0) = 0$$

$$p_k(u) = BF_0(u) \cdot p_{k-1} + BF_1(u) \cdot p_k + BF_2(u) \cdot p_{k+1} + BF_3(u) \cdot p_{k+2}$$
Comparison: Cardinal vs. Cubic B

Cardinal Splines \((s = 1/2)\)

\[
\begin{align*}
BF_0(u) &= -\frac{1}{2} u^3 + u^2 - \frac{1}{2} u \\
BF_1(u) &= \frac{3}{2} u^3 - \frac{5}{2} u^2 + 1 \\
BF_2(u) &= -\frac{3}{2} u^3 + 2u^2 + \frac{1}{2} u \\
BF_3(u) &= \frac{1}{2} u^3 - \frac{1}{2} u^2
\end{align*}
\]

\[
\begin{align*}
BF_0(0) &= 0 & BF_0(1) &= 0 \\
BF_1(0) &= 1 & BF_1(1) &= 0 \\
BF_2(0) &= 0 & BF_2(1) &= 1 \\
BF_3(0) &= 0 & BF_3(1) &= 0
\end{align*}
\]

Cubic B-Splines \((s = 1/2)\)

\[
\begin{align*}
BF_0(u) &= -\frac{1}{2} u^3 + \frac{1}{2} u^2 - \frac{1}{2} u + \frac{1}{6} \\
BF_1(u) &= \frac{1}{2} u^3 - u^2 + \frac{2}{3} \\
BF_2(u) &= -\frac{1}{2} u^3 + \frac{1}{2} u^2 + \frac{1}{2} u + \frac{1}{6} \\
BF_3(u) &= \frac{1}{6} u^3
\end{align*}
\]

\[
\begin{align*}
BF_0(0) &= \frac{1}{6} & BF_0(1) &= 0 \\
BF_1(0) &= \frac{2}{3} & BF_1(1) &= \frac{1}{6} \\
BF_2(0) &= \frac{1}{6} & BF_2(1) &= \frac{2}{3} \\
BF_3(0) &= 0 & BF_3(1) &= \frac{1}{6}
\end{align*}
\]

\[
P_k(u) = BF_0(u) \cdot p_{k-1} + BF_1(u) \cdot p_k + BF_2(u) \cdot p_{k+1} + BF_3(u) \cdot p_{k+2}
\]
Comparison: Cardinal vs. Cubic B

Cardinal Splines \((s = 1/2)\)

\[
\begin{align*}
BF_0'(u) &= -\frac{3}{2}u^2 + 2u - \frac{1}{2} \\
BF_1'(u) &= \frac{9}{2}u^2 - 5u \\
BF_2'(u) &= -\frac{9}{2}u^2 + 4u + \frac{1}{2} \\
BF_3'(u) &= \frac{3}{2}u^2 - u
\end{align*}
\]

\[
\begin{align*}
BF_0'(0) &= -\frac{1}{2} & BF_0'(1) &= 0 \\
BF_1'(0) &= 0 & BF_1'(1) &= -\frac{1}{2} \\
BF_2'(0) &= \frac{1}{2} & BF_2'(1) &= 0 \\
BF_3'(0) &= 0 & BF_3'(1) &= \frac{1}{2}
\end{align*}
\]

Cubic B-Splines \((s = 1/2)\)

\[
\begin{align*}
BF_0'(u) &= -\frac{1}{2}u^2 + u - \frac{1}{2} \\
BF_1'(u) &= \frac{3}{2}u^2 - 2u \\
BF_2'(u) &= -\frac{3}{2}u^2 + u + \frac{1}{2} \\
BF_3'(u) &= \frac{1}{2}u^2
\end{align*}
\]

\[
\begin{align*}
BF_0'(0) &= -\frac{1}{2} & BF_0'(1) &= 0 \\
BF_1'(0) &= 0 & BF_1'(1) &= -\frac{1}{2} \\
BF_2'(0) &= \frac{1}{2} & BF_2'(1) &= 0 \\
BF_3'(0) &= 0 & BF_3''(1) &= \frac{1}{2}
\end{align*}
\]

\[P_k(u) = BF_0(u) \cdot p_{k-1} + BF_1(u) \cdot p_k + BF_2(u) \cdot p_{k+1} + BF_3(u) \cdot p_{k+2}\]
Comparison: Cardinal vs. Cubic B

Cardinal Splines \((s = 1/2)\)

\[
\begin{align*}
BF_0''(u) &= -3u + 2 \\
BF_1''(u) &= 9u - 5 \\
BF_2''(u) &= -9u + 4 \\
BF_3''(u) &= 3u - 1
\end{align*}
\]

\[
\begin{align*}
BF_0''(0) &= 2 & BF_0''(1) &= -1 \\
BF_1''(0) &= -5 & BF_1''(1) &= 4 \\
BF_2''(0) &= 4 & BF_2''(1) &= -5 \\
BF_3''(0) &= -1 & BF_3''(1) &= 2
\end{align*}
\]

Cubic B-Splines \((s = 1/2)\)

\[
\begin{align*}
BF_0''(u) &= -u + 1 \\
BF_1''(u) &= 3u - 2 \\
BF_2''(u) &= -3u + 1 \\
BF_3''(u) &= u
\end{align*}
\]

\[
\begin{align*}
BF_0''(0) &= 1 & BF_0''(1) &= 0 \\
BF_1''(0) &= -2 & BF_1''(1) &= 1 \\
BF_2''(0) &= 1 & BF_2''(1) &= -2 \\
BF_3''(0) &= 0 & BF_3''(1) &= 1
\end{align*}
\]

\[
\mathbf{P}_k(u) = BF_0(u) \cdot \mathbf{p}_{k-1} + BF_1(u) \cdot \mathbf{p}_k + BF_2(u) \cdot \mathbf{p}_{k+1} + BF_3(u) \cdot \mathbf{p}_{k+2}
\]
Comparison: Cardinal vs. Cubic B

Cardinal Splines \((s = 1/2)\)

<table>
<thead>
<tr>
<th>(BF_0'''(u))</th>
<th>(BF_1'''(u))</th>
<th>(BF_2'''(u))</th>
<th>(BF_3'''(u))</th>
</tr>
</thead>
<tbody>
<tr>
<td>(-3)</td>
<td>(9)</td>
<td>(-9)</td>
<td>(3)</td>
</tr>
</tbody>
</table>

P_0(u) = BF_0(u) \cdot p_{k-1} + BF_1(u) \cdot p_k + BF_2(u) \cdot p_{k+1} + BF_3(u) \cdot p_{k+2}

Cubic B-Splines \((s = 1/2)\)

<table>
<thead>
<tr>
<th>(BF_0'''(u))</th>
<th>(BF_1'''(u))</th>
<th>(BF_2'''(u))</th>
<th>(BF_3'''(u))</th>
</tr>
</thead>
<tbody>
<tr>
<td>(-1)</td>
<td>(3)</td>
<td>(-3)</td>
<td>(1)</td>
</tr>
</tbody>
</table>

P_0(u) = BF_0(u) \cdot p_{k-1} + BF_1(u) \cdot p_k + BF_2(u) \cdot p_{k+1} + BF_3(u) \cdot p_{k+2}
Blending Functions

Properties:

• Translation Equivariance:
  \[ BF_0(u) + BF_1(u) + BF_2(u) + BF_3(u) = 1 \] for all \( 0 \leq u \leq 1 \).

• Continuity:
  \[ 0 = BF_0(1), BF_0(0) = BF_1(1), BF_1(0) = BF_2(1), BF_2(0) = BF_3(1), BF_3(0) = 0 \]

• Convex Hull Containment:
  A point is inside the convex hull of a collection of points if and only if it can be expressed as the weighted average of the points, where all the weights are non-negative.
  \[ BF_0(u), BF_1(u), BF_2(u), BF_3(u) \geq 0, \text{ for all } 0 \leq u \leq 1. \]

\[ p_k(u) = BF_0(u) \cdot p_{k-1} + BF_1(u) \cdot p_k + BF_2(u) \cdot p_{k+1} + BF_3(u) \cdot p_{k+2} \]
**Comparison: Cardinal vs. Cubic B**

Cardinal Splines \((s = 1/2)\)

\[
P_k(u) = BF_0(u) \cdot p_{k-1} + BF_1(u)
\]

Cubic B-Splines \((s = 1/2)\)

Note:
We’ve only shown convex hull containment of uniform cubic B-splines for a particular choice of \(s\).
Comparison: Cardinal vs. Cubic B

Cardinal Splines ($s = 1/2$)  

Cubic B-Splines ($s = 1/2$)

\[
\mathbf{P}_k(u) = BF_0(u) \cdot \mathbf{p}_{k-1} + BF_1(u) \cdot \mathbf{p}_k + BF_2(u) \cdot \mathbf{p}_{k+1} + BF_3(u) \cdot \mathbf{p}_{k+2}
\]
Blending Functions

Properties:

• Translation Equivariance:
  \[ BF_0(u) + BF_1(u) + BF_2(u) + BF_3(u) = 1 \text{ for all } 0 \leq u \leq 1. \]

• Continuity:
  
  \[ 0 = BF_0(1), \quad BF_0(0) = BF_1(1), \quad BF_1(0) = BF_2(1), \quad BF_2(0) = BF_3(1), \quad BF_3(0) = 0 \]

• Convex Hull Containment:
  \[ BF_0(u), BF_1(u), BF_2(u), BF_3(u) \geq 0, \text{ for all } 0 \leq u \leq 1. \]

• Interpolation:
  
  We want the spline segments to satisfy:

  \[ P_k(0) = p_k \quad \text{and} \quad P_k(1) = p_{k+1} \]

  \[ \Rightarrow \text{ At the end-points, the blending functions satisfy:} \]

  \[
  \begin{align*}
  BF_0(0) & = 0 & BF_0(1) & = 0 \\
  BF_1(0) & = 1 & BF_1(1) & = 0 \\
  BF_2(0) & = 0 & BF_2(1) & = 1 \\
  BF_3(0) & = 0 & BF_3(1) & = 0 
  \end{align*}
  \]

\[
P_k(u) = BF_0(u) \cdot p_{k-1} + BF_1(u) \cdot p_k + BF_2(u) \cdot p_{k+1} + BF_3(u) \cdot p_{k+2}
\]
Comparison: Cardinal vs. Cubic B

Cardinal Splines ($s = 1/2$)

- $BF_0(u) = -\frac{1}{2}u^3 + u^2 - \frac{1}{2}u$
- $BF_1(u) = \frac{3}{2}u^3 - \frac{5}{2}u^2 + 1$
- $BF_2(u) = -\frac{3}{2}u^3 + 2u^2 + \frac{1}{2}u$
- $BF_3(u) = \frac{1}{2}u^3 - \frac{1}{2}u^2$

- $BF_0(0) = 0$
- $BF_0(1) = 0$
- $BF_1(0) = 1$
- $BF_1(1) = 0$
- $BF_2(0) = 0$
- $BF_2(1) = 1$
- $BF_3(0) = 0$
- $BF_3(1) = 0$

Cubic B-Splines ($s = 1/2$)

- $BF_0(u) = -\frac{1}{6}u^3 + \frac{1}{2}u^2 - \frac{1}{2}u + \frac{1}{6}$
- $BF_1(u) = \frac{1}{2}u^3 - u^2 + \frac{2}{3}$
- $BF_2(u) = -\frac{1}{2}u^3 + \frac{1}{2}u^2 + \frac{1}{2}u + \frac{1}{6}$
- $BF_3(u) = \frac{1}{6}u^3$

- $BF_0(0) = \frac{1}{6}$
- $BF_0(1) = 0$
- $BF_1(0) = \frac{1}{2}$
- $BF_1(1) = \frac{1}{6}$
- $BF_2(0) = \frac{1}{3}$
- $BF_2(1) = \frac{1}{3}$
- $BF_3(0) = 0$
- $BF_3(1) = \frac{1}{6}$

$P_k(u) = BF_0(u) \cdot p_{k-1} + BF_1(u) \cdot p_k + BF_2(u) \cdot p_{k+1} + BF_3(u) \cdot p_{k+2}$
# Blending Functions

## Properties:

- **Translation Equivariance:**
  \[ BF_0(u) + BF_1(u) + BF_2(u) + BF_3(u) = 1 \] for all \( 0 \leq u \leq 1 \).

- **Continuity:**
  \[
  \begin{align*}
  0 &= BF_0(1) \\
  BF_0(0) &= BF_1(1) \\
  BF_1(0) &= BF_2(1) \\
  BF_2(0) &= BF_3(1) \\
  BF_3(0) &= 0
  \end{align*}
  \]

- **Convex Hull Containment:**
  \[ BF_0(u), BF_1(u), BF_2(u), BF_3(u) \geq 0, \text{ for all } 0 \leq u \leq 1. \]

- **Interpolation:**
  \[
  \begin{align*}
  BF_0(0) &= 0 & BF_0(1) &= 0 \\
  BF_1(0) &= 1 \quad \text{and} \quad BF_1(1) &= 0 \\
  BF_2(0) &= 0 & BF_2(1) &= 1 \\
  BF_3(0) &= 0 & BF_3(1) &= 0
  \end{align*}
  \]

\[
P_k(u) = BF_0(u) \cdot p_{k-1} + BF_1(u) \cdot p_k + BF_2(u) \cdot p_{k+1} + BF_3(u) \cdot p_{k+2}
\]
Summary

- A spline is a *piecewise polynomial function* whose derivatives satisfy some *continuity constraints* across curve junctions.

- Looked at specification for 3 splines:
  - Hermite \{Interpolating, cubic, $C^1$\}
  - Cardinal \{Approximating, convex-hull containment, cubic, $C^2$\}
  - Uniform Cubic B-Spline

Spline Demo ($t = 1 - 2s$)