Parametric Curves

Michael Kazhdan

(601.457/657)

HB 10.6 -- 10.9
FvDFH 11.2
Parametric Curves

Given a 1D control lattice

• Compute a smooth curve passing through/near the control points
Parametric Surfaces

Given a 2D control lattice

• Compute a smooth surface passing through/near the control points

Very closely related to subdivision surfaces!

“Exact Evaluation Of Catmull-Clark Subdivision Surfaces At Arbitrary Parameter Values”. [Stam, 1998]
Goals

• Some attributes we would like to have:
  ◦ Local support
  ◦ Simple/predictable
  ◦ Continuous

• We’ll satisfy these goals using:
  ◦ Piecewise
  ◦ Polynomials
What is a Spline in CG?

A spline is a *piecewise polynomial function* whose derivatives satisfy *continuity constraints* across curve boundaries.
What is a Spline in CG?

**Piecewise**: the spline is a collection of parametric curves segments joined together.

**Polynomial functions**: each segment is a parametric polynomial curve.
A parametric curve in $d$-dimensions is defined by a collection of 1D functions of one variable giving the coordinates of points on the curve at each $u$ value:

$$\Phi(u) = (x_1(u), \ldots, x_d(u))$$

$\Phi(u) = (\cos u, \sin u, u)$

Note: A parametric curve is not the graph of a function.

Courtesy of C.K. Shene
Derivatives

If $\Phi(u) = (x(u), y(u))$ is the parametric equation of a curve, the parametric derivative of the curve at a point $u_0$ is the vector:

$$\Phi'(u_0) = (x'(u_0), y'(u_0))$$

which points in a direction tangent to the curve.

**Note:**
The direction of the derivative is determined by the path that the curve traces out.

The magnitude of the parametric derivative is determined by the tracing speed.
Polynomials

A polynomial in the variable $u$ is:

“An algebraic expression written as a sum of constants multiplied by different powers of a variable.”

$$P(u) = a_0 + a_1 \cdot u + a_2 \cdot u^2 + \cdots + a_n \cdot u^n = \sum_{k=0}^{n} a_k \cdot u^k$$

The constant $a_k$ is referred to as the $k$-th coefficient of the polynomial $P$.

A polynomial $P(u)$ has degree $n$ if for all $k > n$, the coefficients of the polynomial satisfy $a_k = 0$. 
A polynomial in the variable \( u \) is:

“An algebraic expression written as a sum of constants multiplied by different powers of a variable.”

\[ P(u) = a_0 + a_1 \cdot u + a_2 \cdot u^2 + \cdots + a_n \cdot u^n = \sum_{k=0}^{n} a_k \cdot u^k \]

A polynomial of degree \( n \) has \( n + 1 \) degrees of freedom.

Knowing \( n + 1 \) pieces of information about a polynomial of degree \( n \) should give enough information to reconstruct the coefficients.
Polynomials (Matrices)

\[ P(u) = a_0 + a_1 \cdot u + a_2 \cdot u^2 + \cdots + a_n \cdot u^n = \sum_{k=0}^{n} a_k \cdot u^k \]

The polynomial \( P \) can be expressed as the matrix multiplication of a row vectors containing the powers of \( u \) and a column vector containing the coefficients:

\[
P(u) = (u^n \quad \cdots \quad u^0) \cdot \begin{pmatrix} a_n \\ \vdots \\ a_0 \end{pmatrix}
\]
Polynomials (1st Derivative Matrices)

\[ P(u) = a_0 + a_1 \cdot u + a_2 \cdot u^2 + \cdots + a_n \cdot u^n = \sum_{k=0}^{n} a_k \cdot u^k \]

The derivative of the polynomial is:

\[ P'(u) = a_1 + 2 \cdot a_2 \cdot u + \cdots + n \cdot a_n \cdot u^{n-1} = \sum_{k=1}^{n} k \cdot a_k \cdot u^{k-1} \]

So the derivative of polynomial \( P \) can be expressed as the matrix multiplication:

\[ P'(u) = \begin{pmatrix} n \cdot u^{n-1} & (n-1) \cdot u^{n-2} & \cdots & 1 & 0 \end{pmatrix} \begin{pmatrix} a_n \\ a_{n-1} \\ \vdots \\ a_1 \\ a_0 \end{pmatrix} \]
Polynomials (Matrices)

Example:

Given the values of $P(u)$ at $n + 1$ different locations:

\[ p_0 = P(u_0), \ldots, p_n = P(u_n) \]

\[ p_0 = (u_0^n \ldots u_0^0) \cdot \begin{pmatrix} \vdots \\ a_n \\ \vdots \\ a_0 \end{pmatrix}, \ldots, p_n = (u_n^n \ldots u_n^0) \cdot \begin{pmatrix} \vdots \\ a_n \\ \vdots \\ a_0 \end{pmatrix} \]

We can stack into one linear system:

\[ \begin{pmatrix} p_0 \\ \vdots \\ p_n \end{pmatrix} = \begin{pmatrix} u_0^n \ldots u_0^0 \\ \vdots \ \ \ \ \ \ \ \ \vdots \\ u_n^n \ldots u_n^0 \end{pmatrix} \begin{pmatrix} \vdots \\ a_n \\ \vdots \\ a_0 \end{pmatrix} \]

\[ P(u) = \sum_{k=0}^{n} a_k \cdot u^k \]
Polynomials (Matrices)

Example:

Given the values of $P(u)$ at $n + 1$ different locations:

$p_0 = P(u_0), \ldots, p_n = P(u_n)$

We can stack into one linear system, and invert:

\[
\begin{pmatrix}
  p_0 \\
  \vdots \\
  p_n
\end{pmatrix} = \begin{pmatrix}
  u_0^n & \cdots & u_0^0 \\
  \vdots & \ddots & \vdots \\
  u_n^n & \cdots & u_n^0
\end{pmatrix} \begin{pmatrix}
  a_n \\
  \vdots \\
  a_0
\end{pmatrix} \Rightarrow \begin{pmatrix}
  a_n \\
  \vdots \\
  a_0
\end{pmatrix} = \begin{pmatrix}
  u_0^n & \cdots & u_0^0 \\
  \vdots & \ddots & \vdots \\
  u_n^n & \cdots & u_n^0
\end{pmatrix}^{-1} \begin{pmatrix}
  p_0 \\
  \vdots \\
  p_n
\end{pmatrix}
\]
Parametric Polynomial Curves

Examples:

\[ x(u) = u \]
\[ y(u) = u \]
\[ x(u) = \frac{u^2}{2} - 2 \]
\[ y(u) = \frac{u^2}{2} - 2 \]
\[ x(u) = \frac{u^3}{2} - 2u \]
\[ y(u) = \frac{u^3}{2} - 2u \]
Parametric Polynomial Curves

Examples:

• When $x(u) = u$, the curve is the graph of $y(u)$.

\begin{align*}
x(u) &= u \\
y(u) &= u \\
\quad &= u^2 - 2 \\
x(u) &= \frac{u^2}{2} - 2 \\
y(u) &= \frac{u^3}{2} - 2u \\
\end{align*}
Parametric Polynomial Curves

Examples:

- When $x(u) = u$, the curve is the graph of $y(u)$.
- Different parametric equations can trace out the same curve.
Parametric Polynomial Curves

Examples:

- When $x(u) = u$, the curve is the graph of $y(u)$.
- Different parametric equations can trace out the same curve.
- As the degree gets larger, the complexity of the curve increases.
Parametric Curves (in $\mathbb{R}^d$)

Goal:

Given a sequence of points, $\{p_1, \cdots, p_m\} \subset \mathbb{R}^d$, define a parametric curve that passes through/near the points
Parametric Curves (in $\mathbb{R}^d$)

Direct Approach:

Solve for the $d \times m$ coefficients of a parametric polynomial curve of degree $m - 1$, passing through the points.

Limitations:

• No local control

• As the number of points increases:
  ◦ The dimension increases and the curve oscillates more
  ◦ Requires inverting a large linear system

Polynomial Fitting Demo
Piecewise parametric polynomials

Approach:

Fit low-order polynomials to groups of points so that the combined curve passes through/near the points.
Piecewise parametric polynomials

Approach:
Fit low-order polynomials to groups of points so that the combined curve passes through/near the points

Properties:
- **Local Control:**
  » Curve segments are local
- **Simplicity**
  » Curve segments are low-order polynomials
- **Continuity/Smoothness**
  » How do we guarantee smoothness?
What is a Spline in CG?

Continuity:

Within the parameterized domain, the polynomial functions are smooth.

The values/derivatives \( P_1(u) \quad u \in [0,1) \), \( P_2(u) \quad u \in [0,1) \), and \( P_3(u) \quad u \in [0,1) \) of the polynomial functions must satisfy continuity constraints across the curve boundaries.

\[
P_i(u) = \sum_{j=0}^{n} a_{ij} \cdot u^j
\]
Continuity/Smoothness

Continuity:

Values/derivatives of the two curves are equal where they meet.

- $C^0$: function is continuous
  \[ P_i(1) = P_{i+1}(0) \]
- $C^1$: function is continuous and 1st derivatives equal
  \[ C^0 \text{ and } P'_i(1) = P'_{i+1}(0) \]
- $C^2$: function is continuous and 1st and 2nd derivatives are equal
  \[ C^1 \text{ and } P''_i(1) = P''_{i+1}(0) \]
- $C^k$: function is continuous and ...
Overview

• What is a Spline?

• Specific Examples:
  ◦ Hermite Splines
  ◦ Cardinal 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.
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

• Let \( P_k(u) = (x_k(u), y_k(u)) \) with \( 0 \leq u \leq 1 \) be the polynomial curve for the section between control points \( \{p_k, \vec{t}_k\} \) and \( \{p_{k+1}, \vec{t}_{k+1}\} \).

• Boundary conditions are:
  - \( P_k(0) = p_k \)
  - \( P_k(1) = p_{k+1} \)
  - \( P_k'(0) = \vec{t}_k \)
  - \( P_k'(1) = \vec{t}_{k+1} \)

• Solve for the coefficients of the polynomials \( x_k(u) \) and \( y_k(u) \) that satisfy the boundary conditions.

Note: Since we have four constraints (per dimension) we need a cubic polynomial.
Specific Example: Hermite Splines

We can express the polynomials:

\[ P_k(u) = a \cdot u^3 + b \cdot u^2 + c \cdot u + d \]

\[ \downarrow \]

\[ P'_k(u) = 3 \cdot a \cdot u^2 + 2 \cdot b \cdot u + c \]

Using the matrix representations:

\[
\begin{pmatrix}
a \\
b \\
c \\
d
\end{pmatrix}
\begin{pmatrix}
u^3 \\
u^2 \\
u \\
1
\end{pmatrix}
\]

\[
\begin{pmatrix}
3 \cdot u^2 \\
2 \cdot u \\
1 \\
0
\end{pmatrix}
\begin{pmatrix}
a \\
b \\
c \\
d
\end{pmatrix}
\]

By abuse of notation, we will think of the coefficients \(a, b, c,\) and \(d\) as \(d\)-dimensional vectors rather than scalars so that \(P_k(u)\) is a function taking values in \(\mathbb{R}^d\).
Specific Example: Hermite Splines

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

The values/derivatives at the end-points are:

\[
p_k = P_k(0) = (0 \ 0 \ 0 \ 1) \cdot \begin{pmatrix} a \\ b \\ c \\ d \end{pmatrix}
\]
Specific Example: Hermite Splines

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

The values/derivatives at the end-points are:

\[ p_k = P_k(0) = (0 \quad 0 \quad 0 \quad 1) \cdot \begin{pmatrix} a \\ b \\ c \\ d \end{pmatrix} \]
\[ p_{k+1} = P_k(1) = (1 \quad 1 \quad 1 \quad 1) \cdot \begin{pmatrix} a \\ b \\ c \\ d \end{pmatrix} \]
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:

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

\[ p_{k+1} = P_k(1) = (1 \quad 1 \quad 1 \quad 1) \cdot \begin{pmatrix} a \\ b \\ c \\ d \end{pmatrix} \]
Specific Example: Hermite Splines

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

The values/derivatives at the end-points are:

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

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

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

We can combine the equations into a single matrix expression:

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

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

Inverting, we get:

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

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

Inverting, we get:

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

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

Using the fact that:

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

We get:

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

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

Multiplying out and rearranging terms, we get:

\[ P_k(u) = p_k(2u^3 - 3u^2 + 1) + p_{k+1}(-2u^3 + 3u^2) + \hat{t}_k(u^3 - 2u^2 + u) + \hat{t}_{k+1}(u^3 - u^2) \]
Specific Example: Hermite Splines

\[ P_k(u) = p_k(2u^3 - 3u^2 + 1) + p_{k+1}(-2u^3 + 3u^2) + \vec{t}_k(u^3 - 2u^2 + u) + \vec{t}_{k+1}(u^3 - u^2) \]

Setting:

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

we can write \( P_k(u) \) as:

\[ P_k(u) = H_0(u) \cdot p_k + H_1(u) \cdot p_{k+1} + H_2(u) \cdot \vec{t}_k + H_3(u) \cdot \vec{t}_{k+1} \]
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

\[
P_k(u) = H_0(u) \cdot \mathbf{p}_k + H_1(u) \cdot \mathbf{p}_{k+1} + H_2(u) \cdot \mathbf{t}_k + H_3(u) \cdot \mathbf{t}_{k+1}
\]
Specific Example: Hermite Splines

Setting:

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

When $u = 0$:

- $H_0(u) = 1$
- $H_1(u) = 0$
- $H_2(u) = 0$
- $H_3(u) = 0$

So $P_k(0) = \mathbf{p}_k$

\[
P_k(u) = H_0(u) \cdot \mathbf{p}_k + H_1(u) \cdot \mathbf{p}_{k+1} + H_2(u) \cdot \mathbf{t}_k + H_3(u) \cdot \mathbf{t}_{k+1}
\]
Specific Example: Hermite Splines

Setting:

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

When $u = 1$:
- $H_0(u) = 0$
- $H_1(u) = 1$
- $H_2(u) = 0$
- $H_3(u) = 0$

So $P_k(1) = p_{k+1}$

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

Setting:

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

When $u = 0$:

- $H_0'(u) = 0$
- $H_1'(u) = 0$
- $H_2'(u) = 1$
- $H_3'(u) = 0$

So $P_k'(0) = \hat{t}_k$

\[
P_k'(u) = H_0'(u) \cdot p_k + H_1'(u) \cdot p_{k+1} + H_2'(u) \cdot \hat{t}_k + H_3'(u) \cdot \hat{t}_{k+1}
\]
Specific Example: Hermite Splines

Setting:

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

When \( u = 1 \):

- \( H_0'(1) = 0 \)
- \( H_1'(1) = 0 \)
- \( H_2'(1) = 0 \)
- \( H_3'(1) = 1 \)

So \( P_k'(1) = \vec{t}_{k+1} \)

\[
P_k'(u) = H_0'(u) \cdot p_k + H_1'(u) \cdot p_{k+1} + H_2'(u) \cdot \vec{t}_k + H_3'(u) \cdot \vec{t}_{k+1}
\]
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
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.
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.

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

• Let $P_k(u) = (x_k(u), y_k(u))$ with $0 \leq u \leq 1$ be the polynomial curve for the section between control points $p_k$ and $p_{k+1}$.

• Boundary conditions are:
  - $P_k(0) = p_k$
  - $P_k(1) = p_{k+1}$
  - $P_k'(0) = s(p_{k+1} - p_{k-1})$
  - $P_k'(1) = s(p_{k+2} - p_k)$

• Solve for the coefficients of the polynomials $x_k(u)$ and $y_k(u)$ that satisfy the boundary conditions.

The parameter $s$ controls looseness versus tightness of curve
Specific Example: Cardinal Splines

Recall:

The Hermite matrix determines the coefficients of the polynomial from the positions and the derivatives of the end-points:

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

We can express the boundary constraints as:

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

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

Setting:

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

Blending Functions

For $s = 1/2$:

$$P_k(u) = C_0(u) \cdot p_{k-1} + C_1(u) \cdot p_k + C_2(u) \cdot p_{k+1} + C_3(u) \cdot p_{k+2}$$