Curve Modeling

- Spline Curve
- Control Points
- Convex Hull
- Shape of spline curve (Interpolation Splines Approximation Splines)
- Continuity Conditions (Parametric & Geometric)
- Linear Spline Interpolation
- Cubic Spline Interpolation
  1. Natural Cubic Splines
  2. Hermit Splines
  3. Cardinal Splines
Spline Curve
Spline Curve

In drafting terminology, a spline is a **flexible strip** used to produce a smooth curve through a designated set of points.
Spline Curve

- Several small weights are distributed along the length of the strip to hold it in position on the drafting table as the curve is drawn.

Weights
Spline Curve

- The term *spline curve* originally referred to a curve drawn in this manner.
Spline Curve

- In **modeling**, the term **spline curve** refers to a **composite** curve formed with **polynomial sections** satisfying specified **continuity conditions** at the boundary of the pieces.
Control Points

- A spline curve is specified by given a set of coordinate positions, called **control points**.
- Control points indicate the general **shape** of the curve.

![Control Points Diagram](image)
Control Points

- A spline curve is defined, modified, and manipulated with operations on the control points.
The convex polygon boundary that enclose a set control points is called the convex hull.
Convex Hull

- The convex hull is a rubber band stretched around the positions of the control points so that each control point is either on the perimeter of the hull or inside it.
Remind:

- A region is **convex** if the line segment joining any two points in the region is also **within** the region.
Shape of the Spline Curve
Shape of the Curve

- Control points are fitted with piecewise continuous parametric polynomial function in one of two way:

1. Interpolation Splines

2. Approximation Splines
**Interpolation Splines**

- When polynomial sections are fitted so that the curve *passes* through each control point, the resulting curve is said to *interpolate* the set of control points.
Interpolation Splines

- Interpolation curves are commonly used to **digitize drawing** or to specify animation **paths**.
Approximation Splines

When polynomial sections are fitted to the general control point path without necessarily passing through any control point, the resulting curve is said to approximate the set if control points.
Approximation Splines

- Approximation curves are used as design tools to structure object surfaces.
Parametric Continuity Conditions
Parametric Continuity Conditions

- To represent a curve as a series of piecewise parametric curves, these curves must fit together reasonably.

...Continuity!

discontinuous
Parametric Continuity Conditions

- Each section of a spline is described with a set of parametric coordinate functions of the form:

\[ x = x(u), \quad y = y(u), \quad z = z(u), \quad u_1 \leq u \leq u_2 \]
Parametric Continuity Conditions

- Let $C_1(u)$ and $C_2(u)$, $0 \leq u \leq 1$ be two parametric Curves.

- We set parametric continuity by matching the parametric derivation of adjoining curve sections at their common boundary.
### Parametric Continuity Conditions

- **Zero order parametric ($C^0$):** Simply the curves meet, $C_1(1) = C_2(0)$.

- **First order parametric ($C^1$):** The first parametric derivations for two successive curve sections are equal at their joining point, $C'_1(1) = C'_2(0)$. 

![Diagram showing different levels of continuity](image)
Parametric Continuity Conditions

- **Second order parametric (C^2):** Both the first and second parametric derivatives of the two curve sections are the same at the intersection, \( C''_1(1) = C''_2(0) \)
Geometric Continuity Conditions
Geometric Continuity Conditions

- **Geometric Continuity Conditions**: Only require parametric derivatives of the two sections to be proportional to each other at their common boundary instead of equal to each other.
Geometric Continuity Conditions

- **Zero order geometric ($G^0$):** Same as $C^0$
  \[ C_1(1) = C_2(0) \]

- **First order geometric ($G^1$):** The parametric first derivatives are proportional at the intersection of two successive sections.
  \[ C'(1) = \alpha C'(0) \]

- **Second order geometric ($G^2$):** Both the first and second parametric derivatives of the two curve sections are proportional at their boundary,
  \[ C''(1) = \alpha C''(0) \]
Parametric & Geometric Continuity

- A curve generated with geometric continuity condition is similar to one generated with parametric continuity, but with slight differences in curve shape.

- With geometric continuity, the curve is pulled toward the section with the greater tangent vector.
Spline Interpolation
Spline Interpolation

- Linear Spline Interpolation
- Cubic Spline Interpolation
Linear Spline Interpolation
Linear Spline Definition

- Given a set of control points, linear interpolation spline is obtained by fitting the input points with a piecewise linear polynomial curve that passes through every control point.
Linear Spline Interpolation

- A linear polynomial with unknowns \( a \) & \( b \):

\[
\begin{align*}
  x(u) &= a_x u + b_x \\
  y(u) &= a_y u + b_y, \quad (0 \leq u \leq 1) \\
  z(u) &= a_z u + b_z \\

  p(u) &= au + b, \quad 0 \leq u \leq 1
\end{align*}
\]
Linear Spline Interpolation

- We need to determine the values of $a$ and $b$ in the linear polynomial for each of the $n$ curve section.

$$p(u) = au + b$$

$$= [u \quad 1] \cdot \begin{bmatrix} a \\ b \end{bmatrix}$$

- Substituting endpoint values 0 and 1 for parameter $u$:

$$\begin{bmatrix} p_0 \\ p_1 \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 1 & 1 \end{bmatrix} \cdot \begin{bmatrix} a \\ b \end{bmatrix}$$
Linear Spline Interpolation

- Solving this equation for the polynomial coefficients:

\[
\begin{bmatrix}
a \\
b
\end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 1 & 1 \end{bmatrix}^{-1} \cdot \begin{bmatrix}
p_0 \\
p_1
\end{bmatrix}
\]

\[
= \begin{bmatrix} -1 & 1 \\ 0 & 1 \end{bmatrix} \cdot \begin{bmatrix}
p_0 \\
p_1
\end{bmatrix}
\]

- \( P(u) \) can be written in terms of the boundary conditions:

\[
p(u) = \begin{bmatrix} u & 1 \end{bmatrix} \cdot \begin{bmatrix}
a \\
b
\end{bmatrix}
\]

\[
p(u) = \begin{bmatrix} u & 1 \end{bmatrix} \cdot \begin{bmatrix} -1 & 1 \\ 0 & 1 \end{bmatrix} \cdot \begin{bmatrix}
p_0 \\
p_1
\end{bmatrix}
\]
Linear Spline Interpolation

The **matrix representation** for linear spline representation:

\[ P(u) = U \cdot M_{spline} \cdot M_{geometry} \]

- **\( M_{geometry} \)** is the matrix containing the **geometry** constraint values (boundary condition) on the spline.

- **\( M_{spline} \)** is the matrix that transform the geometric constraint values to the polynomial coefficients and provides a **characterization** for the spline curve (**Basis Matrix**).
Linear Spline Interpolation

We can expand the matrix representation:

\[ p(u) = \begin{bmatrix} u & 1 \end{bmatrix} \cdot \begin{bmatrix} -1 & 1 \\ 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} p_0 \\ p_1 \end{bmatrix} = \begin{bmatrix} 1 - u & u \end{bmatrix} \cdot \begin{bmatrix} p_0 \\ p_1 \end{bmatrix} \]

Blending Functions

\[ P(u) = P_0 (1 - u) + P_1 (u) = b_0 (u) P_0 + b_1 (u) P_1 \]

\[ P(u) = \sum_{k=0}^{1} P_k b_k (u), \quad 0 \leq u \leq 1 \]
Linear Spline Interpolation

\[ P(u) = P_0(1-u) + P_1(u) \]
\[ = b_0(u)P_0 + b_1(u)P_1 \]

**The Linear spline blending functions:**

<table>
<thead>
<tr>
<th></th>
<th>( B_0(u) )</th>
<th>( B_1(u) )</th>
</tr>
</thead>
<tbody>
<tr>
<td>( u=0 )</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>( u=1 )</td>
<td>0</td>
<td>1</td>
</tr>
</tbody>
</table>

\[ P(u) = \sum_{k=0}^{1} P_k b_k(u), \quad 0 \leq u \leq 1 \]
Linear Spline Interpolation

- The curve is linear combination of the blending functions.
- The piecewise linear interpolation is $C^0$ continuous.
- For a linear polynomial, 2 constraints are required for each segment.

A polynomial of degree $k$ has $k+1$ coefficients and thus requires $k+1$ independent constraints to uniquely determine it.
Linear Spline Applications

- Construction of domes with tubular or truss element.
- Used in some old airplanes built with space truss.
Cubic Spline Interpolation
Cubic Spline Interpolation

**Used to:**

- Set up paths for object motion
- A representation for an existing object or drawing
- Design of objects surface

Cubic splines are more flexible for modeling arbitrary curve shapes such as vehicle or aircraft body.
Cubic Spline Definition:

- Given a set of control points:

\[ P_k = (x_k, y_k, z_k), \quad k = 0, 1, 2, \ldots, n \]

- **Cubic interpolation spline** is obtained by fitting the input points with a piecewise **cubic polynomial** curve that passes through every control points.
Cubic Spline Interpolation

- A cubic polynomial:

\[
x(u) = a_x u^3 + b_x u^2 + c_x u + d_x
\]

\[
y(u) = a_y u^3 + b_y u^2 + c_y u + d_y, \quad (0 \leq u \leq 1)
\]

\[
z(u) = a_z u^3 + b_z u^2 + c_z u + d_z
\]

\[
p(u) = au^3 + bu^2 + cu + d, \quad 0 \leq u \leq 1
\]
Linear Spline Interpolation

\[ p(u) = au^3 + bu^2 + cu + d, \quad 0 \leq u \leq 1 \]

- We need to determine the values of \( a \), \( b \), \( c \) and \( d \) in the cubic polynomial for each of the \( n \) curve section.

- By setting enough boundary conditions at the “joints” between curve sections we can obtain numerical values for all the coefficients.
Cubic Spline Types

Methods for setting the boundary conditions for cubic interpolation splines:

1. Natural Cubic Splines

2. Hermit Splines

3. Cardinal Splines

4. Kochanek-Bartels Splines
Natural Cubic Splines
Natural Cubic Splines

- Natural cubic splines are developed for graphics application.

- Natural cubic splines have $C^2$ continuity.
Natural Cubic Splines

If we have $n+1$ control points to fit, then:

- We have $n$ curve sections
- $4n$ polynomial coefficients to be determined.

3 control points ($n=2$)
2 curve sections
$4 \times 2$ polynomial coefficients to be determined.
Natural Cubic Splines

At each \( n-1 \) interior control points, we have 4 boundary condition:

- The two curve sections on either side of a control point must **first** and **second** parametric derivatives at that control point, and **each curve must pass** through that control point.

At \( P_1 \) interior control:

1. \( C'(1) = C'(0) \)
2. \( C''(1) = C''(0) \)
3. \( C_1(1) = P_1 \)
4. \( C_2(0) = P_1 \)
Natural Cubic Splines

Additional equation:

- First control point $P_0$
- End control points $P_n$

5. $C_1(0) = P_0$
6. $C_2(1) = P_2$
Natural Cubic Splines

We still need two more conditions to be able to determine values for all coefficients.
Natural Cubic Splines

**Method 1:**

Set the second derivatives at $p_0$ and $p_n$ to 0

7. $C''_1(0)=0$

8. $C''_2(1)=0$
Natural Cubic Splines

Method 2:

Add a control point \( P_{-1} \) and a control point \( P_{n+1} \).
Natural Cubic Splines

- Natural cubic splines are a mathematical model for the drafting spline.

Disadvantage:

- **No** “Local Control”: we cannot reconstruct part of the curve without specifying an entirely new set of control points.
Hermit Splines
Hermit Splines

Given a set of control points: \( P_k = (x_k, y_k, z_k), \quad k = 0,1,2,\ldots,n \)

- **Hermit splines** is an interpolating piecewise cubic polynomial with a specified tangent at each control point.

![Diagram of Hermit Splines](image-url)
**Hermit Splines**

- Hermit splines can be adjusted locally because each curve section is only dependent on its endpoint constraints (unlike the natural cubic splines).
Hermit Splines

- **Example:** Change in *Magnitude* of $T_0$
Hermit Splines

Example: Change in Direction of $T_0$
Hermit Splines

- **P(u)** represents a parametric cubic point function for the section between \( p_k \) and \( p_{k+1} \).
  \[
p(u) = au^3 + bu^2 + cu + d, \quad 0 \leq u \leq 1
  \]

- The **boundary conditions** are:
  
  \[
  \begin{align*}
  p(0) &= p_k \\
  p(1) &= p_{k+1} \\
  p'(0) &= Dp_k \\
  p'(1) &= Dp_{k+1}
  \end{align*}
  \]

- **Dp_k** and **Dp_{k+1}** are the values for the parametric derivations (slope of the curve) at control points **P_k** and **P_{k+1}**.
Hermit Splines

\[ p(u) = au^3 + bu^2 + cu + d, \quad 0 \leq u \leq 1 \]

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

The derivative of the point function:

\[ p'(u) = \begin{bmatrix} 3u^2 & 2u & 1 & 0 \end{bmatrix} \cdot \begin{bmatrix} a \\ b \\ c \\ d \end{bmatrix} \]

Boundary conditions

\[ p(0) = p_k \]
\[ p(1) = p_{k+1} \]
\[ p'(0) = Dp_k \]
\[ p'(1) = Dp_{k+1} \]

Constraints in a matrix form:

\[
\begin{bmatrix}
    p_k \\
    p_{k+1} \\
    Dp_k \\
    Dp_{k+1}
\end{bmatrix} =
\begin{bmatrix}
    0 & 0 & 0 & 1 \\
    1 & 1 & 1 & 1 \\
    0 & 0 & 1 & 0 \\
    3 & 2 & 1 & 0
\end{bmatrix}
\begin{bmatrix}
    a \\
    b \\
    c \\
    d
\end{bmatrix}
\]
Solving this equation for the polynomial coefficients:

\[
\begin{bmatrix}
    a \\
    b \\
    c \\
    d
\end{bmatrix} =
\begin{bmatrix}
    0 & 0 & 0 & 1 \\
    1 & 1 & 1 & 1 \\
    0 & 0 & 1 & 0 \\
    3 & 2 & 1 & 0
\end{bmatrix}^{-1}
\begin{bmatrix}
    p_k \\
    p_{k+1} \\
    Dp_k \\
    Dp_{k+1}
\end{bmatrix}
= 
\begin{bmatrix}
    0 & 0 & 0 & 1 \\
    1 & 1 & 1 & 1 \\
    0 & 0 & 1 & 0 \\
    3 & 2 & 1 & 0
\end{bmatrix}
\begin{bmatrix}
    2 & -2 & 1 & 1 \\
    -3 & 3 & -2 & -1 \\
    0 & 0 & 1 & 0 \\
    1 & 0 & 0 & 0
\end{bmatrix}
\begin{bmatrix}
    p_k \\
    p_{k+1} \\
    Dp_k \\
    Dp_{k+1}
\end{bmatrix}
= 
M_H \cdot \begin{bmatrix}
    p_k \\
    p_{k+1} \\
    Dp_k \\
    Dp_{k+1}
\end{bmatrix}
\]

\(M_H\), the Hermit matrix, is the inverse of the boundary constraint matrix.
\[ p(u) = \begin{bmatrix} u^3 & u^2 & u & 1 \end{bmatrix} \cdot \mathbf{a} \]

\[ p(u) = \begin{bmatrix} u^3 & u^2 & u & 1 \end{bmatrix} \cdot \mathbf{M}_H \cdot \begin{bmatrix} p_k \\ p_{k+1} \\ Dp_k \\ Dp_{k+1} \end{bmatrix} \]

\[ p(u) = \begin{bmatrix} 2u^3 - 3u^2 + 1 & -2u^3 + 3u^2 & u^3 - 2u^2 + u & u^3 - u^2 \end{bmatrix} \cdot \begin{bmatrix} p_k \\ p_{k+1} \\ Dp_k \\ Dp_{k+1} \end{bmatrix} \]

**Blending Functions**

\[ P(u) = p_k H_0(u) + p_{k+1} H_1(u) + Dp_k H_2(u) Dp_{k+1} H_3(u) \]
Hermit Splines
Hermit Blending Functions
Hermit Splines

- Hermit polynomials can be useful for some digitizing applications where it may not be too difficult to specify or approximate the curve slopes.

But...

For most problems in modeling, it is more useful to generate spline curve without requiring input values for curve slopes.
Cardinal Splines
Cardinal Splines

- **Cardinal splines** are interpolating piecewise cubic with specified endpoint tangents at the boundary of each curve section.

- The difference is that we **do not have to give the values for the endpoint tangents**.

For a cardinal Spline, the value for the slope at a control point is approximately calculated from the coordinates of the two adjacent control points.
Cardinal Splines

- A cardinal spline section is completely specified with four consecutive control points.
- The middle two control points are the section endpoints, and the other two points are used in the calculation of the endpoint slopes.
Cardinal Splines

The boundary conditions:

\[ P(0) = p_k \]
\[ P(1) = p_{k+1} \]
\[ P'(0) = \frac{1}{2} (1 - t)(p_{k+1} - p_{k-1}) \]
\[ P'(1) = \frac{1}{2} (1 - t)(p_{k+2} - p_k) \]

Thus, the slopes at control points \( p_k \) and \( p_{k+1} \) are taken to be proportional to the chords \( p_{k-1} p_{k+1} \) and \( p_k p_{k+2} \).
Cardinal Splines

The boundary conditions:

\[ P(0) = p_k \]
\[ P(1) = p_{k+1} \]
\[ P'(0) = \frac{1}{2} (1 - t)(p_{k+1} - p_{k-1}) \]
\[ P'(1) = \frac{1}{2} (1 - t)(p_{k+2} - p_k) \]
Cardinal Splines

- Parameter $t$ is called the **tension parameter**.

- Tension parameter controls how loosely or tightly the cardinal spline fits the input control points.

Mathematical expressions:

- $P(0) = p_k$
- $P(1) = p_{k+1}$
- $P'(0) = \frac{1}{2} (1 - t)(p_{k+1} - p_{k-1})$
- $P'(1) = \frac{1}{2} (1 - t)(p_{k+2} - p_k)$
Cardinal Splines

- When $t=0$ is called the Catmull-Rom splines, or Overhauser splines.
Cardinal Splines

\[
P(u) = \begin{bmatrix} u^3 & u^2 & u & 1 \end{bmatrix} \cdot M_C \cdot \begin{bmatrix} p_{k-1} \\ p_k \\ p_{k+1} \\ p_{k+2} \end{bmatrix}
\]

\[
P(0) = p_k
\]
\[
P(1) = p_{k+1}
\]
\[
P'(0) = \frac{1}{2} (1 - t)(p_{k+1} - p_{k-1})
\]
\[
P'(1) = \frac{1}{2} (1 - t)(p_{k+2} - p_k)
\]

\[
M_C = \begin{bmatrix} -s & 2 - s & s - 2 & s \\ 2s & s - 3 & 3 - 2s & -s \\ -s & 0 & s & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}
\]

\[
s = (1 - t)/2
\]
Cardinal Splines

\[ P(0) = p_k \]
\[ P(1) = p_{k+1} \]
\[ P'(0) = \frac{1}{2} (1 - t)(p_{k+1} - p_{k-1}) \]
\[ P'(1) = \frac{1}{2} (1 - t)(p_{k+2} - p_k) \]

\[ P(u) = p_{k-1}(-su^3 + 2su^2 - su) + p_k[(2 - s)u^3 + (s - 3)u^2 + 1] + p_{k+1}[(s - 2)u^3 + (3 - 2s)u^2 + su] + p_{k+2}(su^3 - su^2) \]

\[ = p_{k-1} \text{CAR}_0(u) + p_k \text{CAR}_1(u) + p_{k+1} \text{CAR}_2(u) + p_{k+2} \text{CAR}_3(u) \]

- The polynomials \( \text{CAR}_k(u) \) for \( k=0,1,2,3 \) are the **cardinal functions**.
Cardinal Splines
Cardinal functions

\[ C_0, C_1, C_2, C_3 \]

\[ S=1 \]

\[ S=2 \]
Homework?
بدنه هواپیما و اسپلاین های معرفی شده بیشتر شباهت دارد. 

را در نظر بگیرید. در مقطعی از بدنه که مجاور بال و محل تقطع بال با بدنه است؛ بررسی کنید شکل مقطع به کدام یک از اسپلاین های معرفی شده بیشتر شباهت دارد.