Lecture Notes #9 - Curves

Reading:

- Angel: Chapter 9
- Foley et al., Sections 11(intro) and 11.2

Overview

- Introduction to mathematical splines
- Bezier curves
- Continuity conditions (C^0, C^1, C^2, G^1, G^2)
- Creating continuous splines
  - C^2 interpolating splines
  - B-splines
  - Catmull-Rom splines

Introduction

Mathematical splines are motivated by the "loftsman's spline":

- Long, narrow strip of wood or plastic
- Used to fit curves through specified data points
- Shaped by lead weights called "ducks"
- Gives curves that are "smooth" or "fair"

Such splines have been used for designing:

- Automobiles
- Ship hulls
- Aircraft fuselages and wings

Requirements

Here are some requirements we might like to have in our mathematical splines:

- Predictable control
- Multiple values
- Local control
- Versatility
- Continuity

Mathematical splines

The mathematical splines we'll use are:

- Piecewise
- Parametric
- Polynomials

Let's look at each of these terms......

Parametric curves

In general, a "parametric" curve in the plane is expressed as:

$$x = x(\mathrm{t})$$

$$y = y(t)$$

<u>Example:</u> A circle with radius r centered at the origin is given by:

$$x = r \cos t$$

$$y = r \sin t$$

By contrast, an "implicit" representation of the circle is:

Parametric polynomial curves

A parametric "polynomial" curve is a parametric curve where each function x(t), y(t) is described by a polynomial:

$$x(t) = \sum_{i=0}^{n} a_i t^i$$

$$y(t) = \sum_{i=0}^{n} b_i t^i$$

Polynomial curves have certain advantages:

- Easy to compute
- Infinitely differentiable

Piecewise parametric polynomial curves

A "piecewise" parametric polynomial curve uses different polynomial functions for different parts of the curve.

- **Advantage:** Provides flexibility
- **Problem:** How do you guarantee smoothness at the joints? (Problem known as "continuity.")

In the rest of this lecture, we'll look at:

1. Bezier curves -- general class of polynomial curves
2. Splines -- ways of putting these curves together

Bezier curves

- Developed simultaneously by Bezier (at Renault) and deCasteljau (at Citroen), circa 1960.
- The Bezier curve $Q(u)$ is defined by nested interpolation:

- V_i's are "control points"
- $\{V_0, \ldots, V_n\}$ is the "control polygon"

Bezier curves: Basic properties

Bezier curves enjoy some nice properties:

- Endpoint interpolation:

$$Q(0) = V_0$$

$$Q(1) = V_n$$

- Convex hull: The curve is contained in the convex hull of its control polygon

- Symmetry:

$$Q(u) \text{ defined by } \{V_0, ..., V_n\}$$

$$\equiv Q(1 - u) \text{ defined by } \{V_n, ... , V_0\}$$

Bezier curves: Explicit formulation

Let's give V_i a superscript V_i^j to indicate the level of nesting.

An explicit formulation for $Q(u)$ is given by the recurrence:

$$V_i^j = (1 - u)\, V_i^{j-1} + u V_{i+1}^{j-1}$$

Explicit formulation, cont.

For $n = 2$, we have:

$Q(u) = V_0^2$

$= (1 - u)V_0^1 + uV_1^1$

$= (1 - u)\,[(1 - u)\,V_0^0 + uV_1^0] + [(1 - u)\,V_1^0 + uV_2^0]$

$= (1 - u)^2V_0^0 + 2u(1 - u)V_1^0 + u^2V_2^0$

In general:

$$Q(u) = \sum_{i=0}^{n} V_i \underbrace{\binom{n}{i} u^i (1-u)^{n-i}}_{B_i^n(u)}$$

$B_i^n(u)$ is the i'th Bernstein polynomial of degree n.

Bezier curves: More properties

Here are some more properties of Bezier curves

$$Q(u) = \sum_{i=0}^{n} V_i \binom{n}{i} u^i (1-u)^{n-i}$$

- Degree: $Q(u)$ is a polynomial of degree n
- Control points: How many conditions must we specify to uniquely determine a Bezier curve of degree n?

More properties, cont.

- Tangents:

$$Q'(0) = n(V_1 - V_0)$$

$$Q'(1) = n(V_n - V_{n-1})$$

- *k*'th derivatives: In general,
  - $Q^{(k)}(0)$ depends only on $V_0, ..., V_k$
  - $Q^{(k)}(1)$ depends only on $V_n, ..., V_{n-k}$
  - (At intermediate points $u \in (0, 1)$, all control points are involved for every derivative.)

Cubic curves

For the rest of this discussion, we'll restrict ourselves to piecewise cubic curves.

- In CAGD, higher-order curves are often used
  - Gives more freedom in design
  - Can provide higher degree of continuity between pieces
- For Graphics, piecewise cubic let's you do just about anything
  - Lowest degree for specifiying points to interpolate and tangents
  - Lowest degree for specifying curve in space

All the ideas here generalize to higher-order curves

Matrix form of Bezier curves

Bezier curves can also be described in matrix form:

$$Q(u) = \sum_{i=0}^{3} V_i \binom{3}{i} u^i (1-u)^{3-i}$$

$$= (1 - u)^3 V_0 + 3u\,(1 - u)^2 V_1 + 3u^2 (1 - u)\, V_2 + u^3 V_3$$

$$= \begin{pmatrix} u^3 & u^2 & u & 1 \end{pmatrix} \begin{pmatrix} -1 & 3 & -3 & 1 \\ 3 & -6 & 3 & 0 \\ -3 & 3 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} V_0 \\ V_1 \\ V_2 \\ V_3 \end{pmatrix}$$

$$= \begin{pmatrix} u^3 & u^2 & u & 1 \end{pmatrix} M_{\text{Bezier}} \begin{pmatrix} V_0 \\ V_1 \\ V_2 \\ V_3 \end{pmatrix}$$

Display: Recursive subdivision

Q: Suppose you wanted to draw one of these Bezier curves -- how would you do it?

A: Recursive subdivision:

Display, cont.

Here's pseudocode for the recursive subdivision display algorithm:

> **procedure** *Display*($\{V_0, ..., V_n\}$):
>
>   **if** $\{V_0, ..., V_n\}$ flat within ε **then**
>
>   Output line segment V_0V_n
>
>   **else**
>
>   Subdivide to produce $\{L_0, ..., L_n\}$ and $\{R_0, ..., R_n\}$
>
>   *Display*($\{L_0, ..., L_n\}$)
>
>   *Display*($\{R_0, ..., R_n\}$)
>
>   **end if**
>
> **end procedure**

Splines

To build up more complex curves, we can piece together different Bezier curves to make "splines."

For example, we can get:

- <u>Positional (C^0) continuity:</u>

- <u>Derivative (C^1) continuity:</u>

Q: How would you build an interactive system to satisfy these constraints?

Advantages of splines

Advantages of splines over higher-order Bezier curves:

- Numerically more stable
- Easier to compute
- Fewer bumps and wiggles

Tangent (G^1) continuity

Q: Suppose the tangents were in opposite directions but not of same magnitude -- how does the curve appear?

This construction gives "tangent (G^1) continuity."

Q: How is G^1 continuity different from C^1?

Curvature (C^2) continuity

Q: Suppose you want even higher degrees of continuity -- e.g., not just slopes but curvatures -- what additional geometric constraints are imposed?

We'll begin by developing some more mathematics.....

Operator calculus

Let's use a tool known as "operator calculus."

Define the operator D by:

$$\mathsf{D}V_i \equiv V_{i+1}$$

Rewriting our explicit formulation in this notation gives:

$$Q(u) = \sum_{i=0}^{n} \binom{n}{i} u^i (1-u)^{n-i} V_i$$

$$= \sum_{i=0}^{n} \binom{n}{i} u^i (1-u)^{n-i} \mathsf{D}_i V_0$$

$$= \sum_{i=0}^{n} \binom{n}{i} (u\mathsf{D})^i (1-u)^{n-i} V_0$$

Applying the binomial theorem gives: $= (u\mathsf{D} + (1 - u))^n V_0$

Taking the derivative

One advantage of this form is that now we can take the derivative:

$$Q'(u) = n(u\Delta + (1 - u))^{n-1} (\Delta - 1) V_0$$

What's $(\Delta - 1) V_0$?

Plugging in and expanding:

$$Q'(u) = n \sum_{i=0}^{n-1} \binom{n-1}{i} u^i (1-u)^{n-1-i} \Delta_i (V_0 - V_1)$$

This gives us a general expression for the derivative $Q'(u)$.

Specializing to n = 3

What's the derivative $Q'(u)$ for a cubic Bezier curve?

Note that:

- When $u = 0$: $Q'(u) = 3(V_1 - V_0)$
- When $u = 1$: $Q'(u) = 3(V_3 - V_2)$

<u>Geometric interpretation:</u>

So for $C1$ continuity, we need to set:

$$3(V_3 - V_2) = 3(W_1 - W_0)$$

Taking the second derivative

Taking the derivative once again yields:

$$Q''(u) = n\,(n - 1)\,(u\mathsf{D} + (1 - u))^{n-2}\,(\mathsf{D} - 1)^2\,V_0$$

What does $(\mathsf{D} - 1)^2$ do?

Second-order continuity

So the conditions for second-order continuity are:

$$(V_3 - V_2) = (W_1 - W_0)$$

$$(V_3 - V_2) - (V_2 - V_1) = (W_2 - W_1) - (W_1 - W_0)$$

Putting these together gives:

Geometric interpretation

C^3 continuity

Summary of continuity conditions

- C^0 straightforward, but generally not enough
- C^3 is too constrained (with cubics)

Creating continuous splines

We'll look at three ways to specify splines with C^1 and C^2 continuity:

1. C^2 interpolating splines
2. B-splines
3. Catmull-Rom splines

C^2 Interpolating splines

The control points specified by the user, called "joints," are <u>interpolated</u> by the spline.

For each of x and y, we needed to specify ______ conditions for each cubic Bezier segment.

So if there are m segments, we'll need ______ constraints.

Q: How many of these constraints are determined by each joint?

In-depth analysis, cont.

At each interior joint *j*, we have:

1. Last curve ends at *j*
2. Next curve begins at *j*
3. Tangents of two curves at *j* are equal
4. Curvature of two curves at *j* are equal

The *m* segments give:

- ______ interior joints
- ______ conditions

The 2 end joints give 2 further contraints:

1. First curve begins at first joint
2. Last curve ends at last joint

Gives _______ constraints altogether.

End conditions

The analysis shows that specifying $m + 1$ joints for m segments leaves 2 extra degrees of freedom.

These 2 extra constraints can be specified in a variety of ways:

- An interactive system
  - Constraints specified as _________
- "Natural" cubic splines
  - Second derivatives at endpoints defined to be 0
- Maximal continuity
  - Require C^3 continuity between first and last pairs of curves

C^2 Interpolating splines

Problem: Describe an interactive system for specifiying C2 interpolating splines.

Solution:

1. Let user specify first four Bezier control points.
2. This constrains next _____ control points -- draw these in.
3. User then picks _____ more
4. Repeat steps 2-3.

Global vs. local control

These C^2 interpolating splines yield only "global control" -- moving any one joint (or control point) changes the entire curve!

Global control is problematic:

- Makes splines difficult to design
- Makes incremental display inefficient

There's a fix, but nothing comes for free. Two choices:

- B-splines
  - Keep C^2 continuity
  - Give up interpolation
- Catmull-Rom splines
  - Keep interpolation
  - Give up C^2 continuity -- provides C^1 only

B-splines

Previous construction (C^2 interpolating splines):

- Choose joints, constrained by the "A-frames."

New construction (B-splines):

- Choose points on A-frames
- Let these determine the rest of Bezier control points and joints

The B-splines I'll describe are known more precisely as "uniform B-splines."

B-spline construction

The points specified by the user in this construction are called "de Boor points."

B-spline properties

Here are some properties of B-splines:

- C^2 continuity
- Approximating
    - Does not interpolate deBoor points
- Locality
    - Each segment determined by 4 deBoor points
    - Each deBoor point determines 4 segments
- Convex hull
    - Curve lies inside convex hull of deBoor points

Algebraic construction of B-splines

$$V_1 = \_\_\_\_\_\_ B_1 + \_\_\_\_\_\_ B_2$$

$$V_2 = \_\_\_\_\_\_ B_1 + \_\_\_\_\_\_ B_2$$

$$V_0 = \_\_\_\_\_\_ [\_\_\_\_\_\_ B_0 + \_\_\_\_\_\_ B_1] + \_\_\_\_\_\_ [\_\_\_\_\_\_ B_1 + \_\_\_\_\_\_ B_2]$$

$$= \_\_\_\_\_\_ B_0 + \_\_\_\_\_\_ B_1 + \_\_\_\_\_\_ B_2$$

$$V_3 = \_\_\_\_\_\_ B_1 + \_\_\_\_\_\_ B_2 + \_\_\_\_\_\_ B_3$$

Algebraic construction of B-splines, cont.

Once again, this construction can be expressed in terms of a matrix:

$$\begin{pmatrix} V_0 \\ V_1 \\ V_2 \\ V_3 \end{pmatrix} = \frac{1}{6} \begin{pmatrix} 1 & 4 & 1 & 0 \\ 0 & 4 & 2 & 0 \\ 0 & 2 & 4 & 0 \\ 0 & 1 & 4 & 1 \end{pmatrix} \begin{pmatrix} B_0 \\ B_1 \\ B_2 \\ B_3 \end{pmatrix}$$

Drawing B-splines

Drawing B-splines is therefore quite simple:

procedure *Draw-B-Spline* ($\{B_0, ..., B_n\}$):

  for i = 0 to n - 3 do

    Convert $B_i, ..., B_{i+3}$ into a Bezier control polygon $V_0, ..., V_3$

    Display ($\{V_0, ... , V_3\}$)

  end for

end procedure

Multiple vertices

Q: What happens if you put more than one control point in the same place?

Some possibilities:

- Triple vertex
- Double vertex
- Collinear vertices

End conditions

You can also use multiple vertices at the endpoints:

- Double endpoint
  - Curve tangent to line between first distinct points
- Triple endpoint
  - Curve interpolates endpoint
  - Starts out with a line segment
- Phantom vertices
  - Gives interpolation without line segment at ends

Catmull-Rom splines

The Catmull-Rom splines

- Give up C^2 continuity
- Keep interpolation

For the derivation, let's go back to the interpolation algorithm. We had 4 conditions at each joint *j*:

1. Last curve ends at *j*
2. Next curve begins at *j*
3. Tangents of two curves at *j* are equal
4. Curvature of two curves at *j* are equal

If we ...

- Eliminate condition 4
- Make condition 3 depend only on local control points

... then we can have <u>local control</u>!

Derivation of Catmull-Rom splines

<u>Idea</u>: (Same as B-splines)

- Start with joints to interpolate
- Build a cubic Bezier curve between successive points

The endpoints of the cubic Bezier are obvious:

$$V_0 = B_1$$
$$V_3 = B_2$$

Q: What should we do for the other two points?

Derivation of Catmull-Rom, cont.

A: Catmull & Rom use *half the magnitude of the vector between adjacent control points*:

Many other choices work -- for example, using an arbitrary constant τ times this vector gives a "tension" control.

Matrix formulation

The Catmull-Rom splines also admit a matrix formulation:

$$\begin{pmatrix} V_0 \\ V_1 \\ V_2 \\ V_3 \end{pmatrix} = \frac{1}{6} \begin{pmatrix} 0 & 6 & 0 & 0 \\ -1 & 6 & 1 & 0 \\ 0 & 1 & 6 & -1 \\ 0 & 0 & 6 & 0 \end{pmatrix} \begin{pmatrix} B_0 \\ B_1 \\ B_2 \\ B_3 \end{pmatrix}$$

<u>Exercise:</u> Derive this matrix.

Properties

Here are some properties of Catmull-Rom splines:

- C^1 Continuity
- Interpolating
- Locality
- **No** convex hull property
    - (Proof left as an exercise.)

(Spline, Bezier, B-Spline)

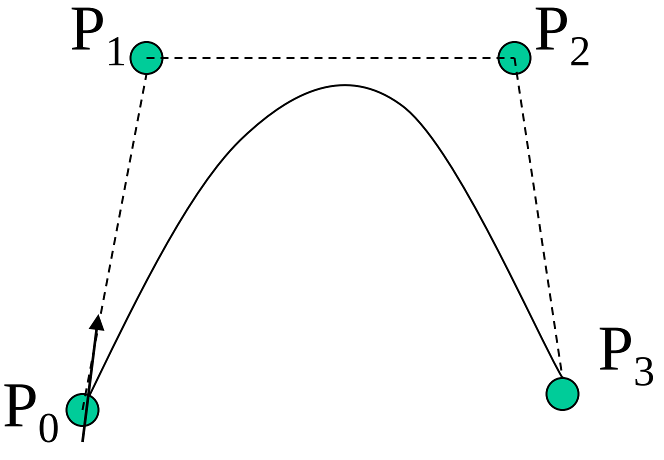
omprakash@teachers.org

Spline

- Drafting terminology
  - Spline is a flexible strip that is easily flexed to pass through a series of design points (control points) to produce a smooth curve.
- Spline curve – a piecewise polynomial (cubic) curve whose first and second derivatives are continuous across the various curve sections.

Bezier curve

- Developed by Paul de Casteljau (1959) and independently by Pierre Bezier (1962).
- French automobil company – Citroen & Renault.



Parametric function

- $P(u) = \sum_{i=0}^{n} B_{n,i}(u)p_i$

Where

$$B_{n,i}(u) = \frac{n!}{i!(n-i)!} u^i(1-u)^{n-i} \qquad 0 <= u <= 1$$

For 3 control points, n = 2

$P(u) = (1-u)^2 p_0 + 2u(1-u)\, p_1 + u^2 p_2$

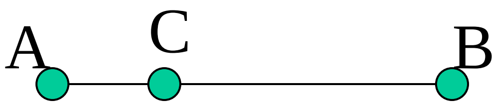
For four control points, n = 3

$P(u) = (1-u)^3 p_0 + 3u(1-u)^2\, p_1 + 3u^2\, (1-u)p_2 + u^3 p_3$

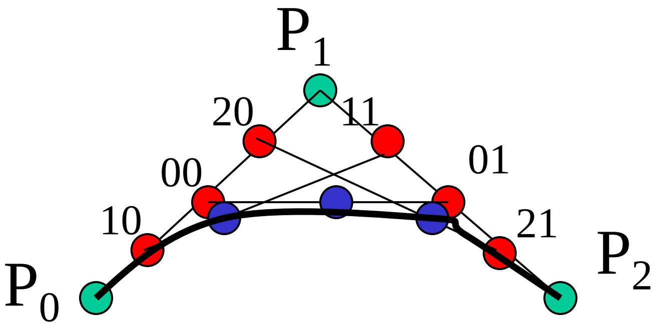
omprakash@teachers.org

algorithm

- De Casteljau
  - Basic concept



    - To choose a point C in line segment AB such that C divides the line segment AB in a ratio of u: 1-u



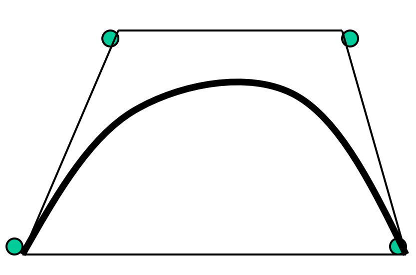
Let u = 0.5
u=0.25
u=0.75

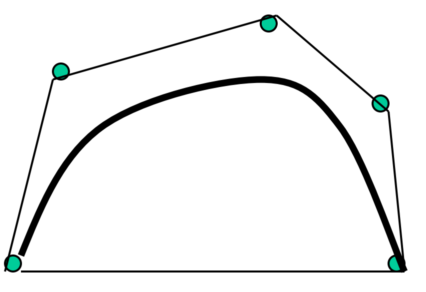
properties

- The curve passes through the first, P_0 and last vertex points, P_n .
- The tangent vector at the starting point P_0 must be given by $P_1 – P_0$ and the tangent P_n given by $P_n – P_{n-1}$
- This requirement is generalized for higher derivatives at the curve's end points. E.g 2nd derivative at P_0 can be determined by P_0 ,P_1 ,P_2 (to satisfy continuity)
- The same curve is generated when the order of the control points is reversed

Properties (continued)

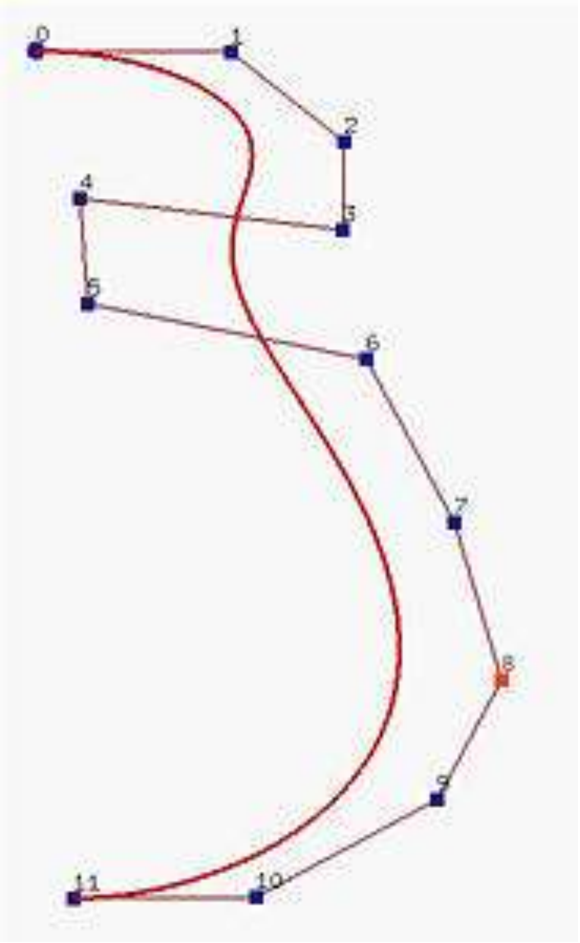
- Convex hull
  - Convex polygon formed by connecting the control points of the curve.
  - Curve resides completely inside its convex hull





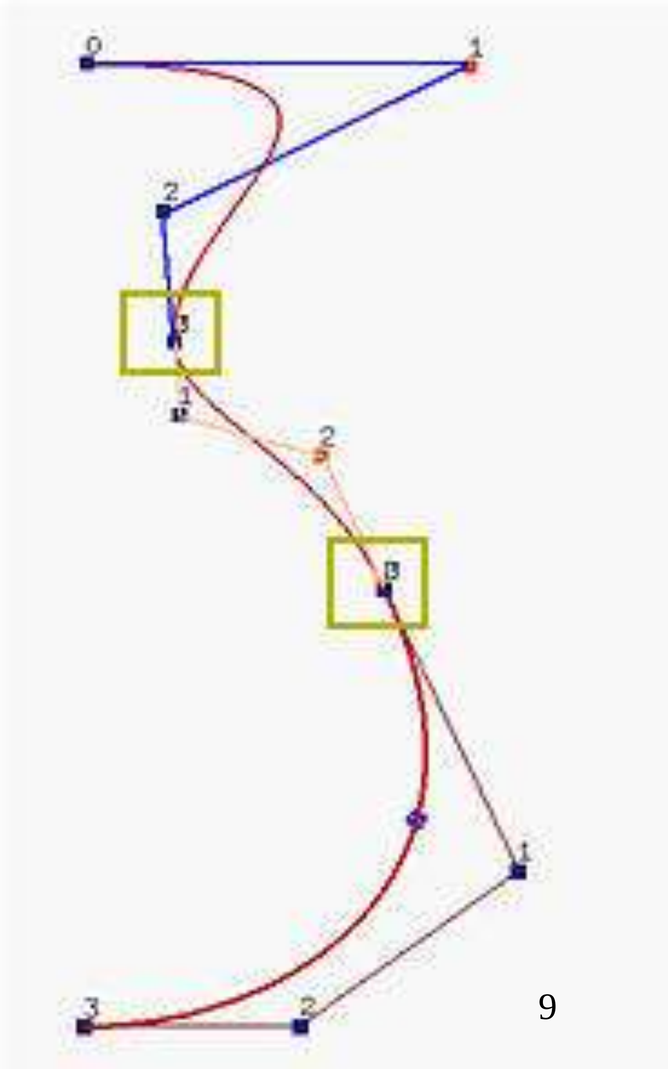
B-Spline

- Motivation (recall bezier curve)
  - The degree of a Bezier Curve is determined by the number of control points
  - E. g. (bezier curve degree 11) – difficult to bend the "neck" toward the line segment $\mathbf{P}_4\mathbf{P}_5$.
  - Of course, we can add more control points.
  - BUT this will increase the degree of the curve → increase computational burden



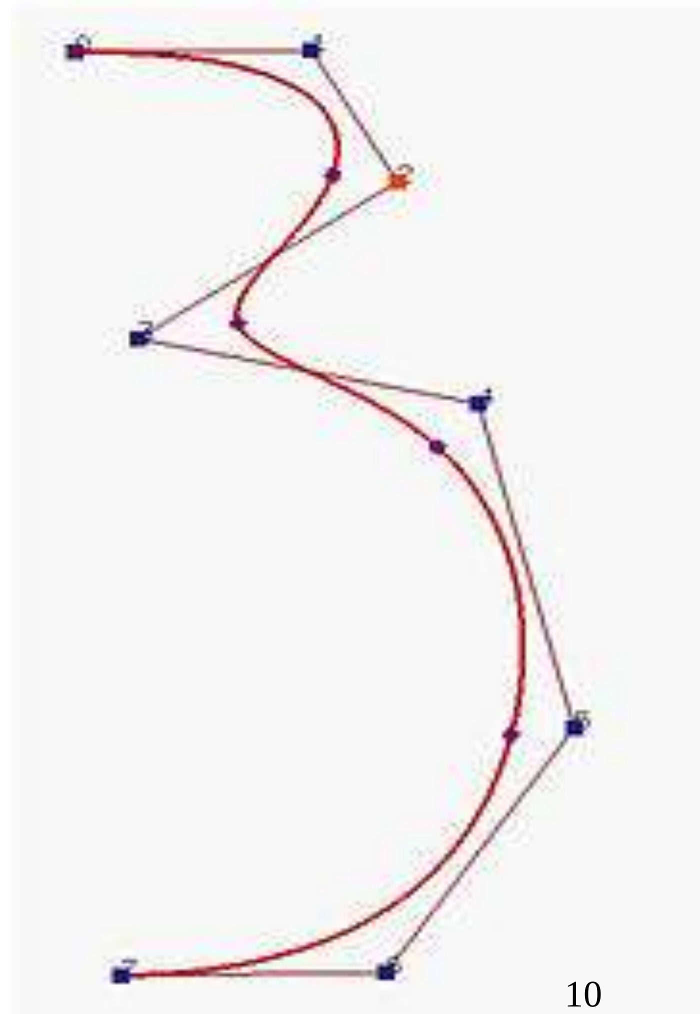
B-Spline

- Motivation (recall bezier curve)
  - Joint many bezier curves of lower degree together (right figure)
  - BUT maintaining continuity in the derivatives of the desired order at the connection point is not easy or may be tedious and undesirable.



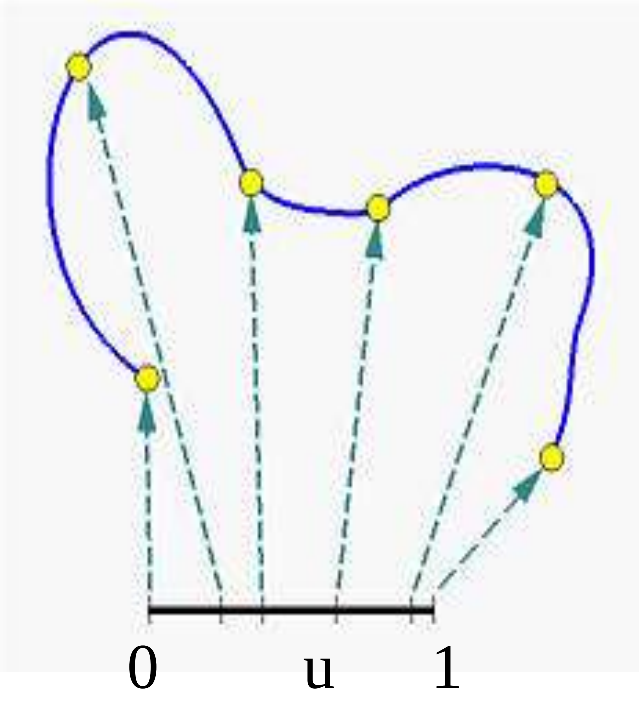
B-Spline

- Motivation (recall bezier curve)
  – moving a control point affects the shape of the entire curve- (*global modification property*) – undesirable.
  - Thus, the solution is B-Spline – the degree of the curve is independent of the number of control points
  - E.g - right figure – a B-spline curve of degree 3 defined by 8 control points



B-Spline

- In fact, there are five Bézier curve segments of degree 3 joining together to form the B-spline curve defined by the control points
- little dots subdivide the B-spline curve into Bézier curve segments.
- Subdividing the curve directly is difficult to do → so, subdivide the domain of the curve by points called *knots*



B-Spline

- In summary, to design a B-spline curve, we need a set of control points, a set of knots and a degree of curve.

B-Spline curve

- $P(u) = \sum_{i=0}^{n} N_{i,k}(u)p_i \quad (u_{min} \leq u \leq u_{max})$.. (1.0)

Where basis function = $N_{i,k}(u)$

Degree of curve → k-1

Control points, p_i → $0 \leq i \leq n$

Knot, u → $u_{min} \leq u \leq u_{max}$

max = n + k

$2 \leq k \leq n+1$

B-Spline : definition

- $P(u) = \sum N_{i,k}(u)p_i$ $(u_{min} \leq u \leq u_m)$
- u_i → knot
- $[u_i, u_{i+1})$ → knot span
- $(u_0, u_1, u_2, \ldots. u_m)$ → knot vector
- The point on the curve that corresponds to a knot u_i, → knot point ,$P(u_i)$
- If knots are equally space → uniform
- If knots are not equally space → non uniform

B-Spline : definition

- Uniform knot vector
  - Individual knot value is **evenly spaced**
  - (0, 1, 2, 3, 4)
  - (0, 0.2, 0.4, 0.6…)
  - Then, normalized to the range [0, 1]
  - (0, 0.25, 0.5, 0.75, 1.0)
  - (0.0,0.1,0.2,0.3,0.4,0.5,0.6,0.7,0.8,0.9,1.0)

B-Spline : definition

- Non-Uniform knot vector
  - Individual knot value is not **evenly spaced**
  - (0, 1, 3, 7, 8)
  - (0, 0.2, 0.3, 0.7……)
  - (0, 0.1, 0.3, 0.4, 0.8 …)
  - Then, normalized to the range [0, 1]
  - (0, 0.15, 0.20, 0.35, 0.40,0.75,0.85,1.0)

Type of B-Spline uniform knot vector

Non-periodic knots (open knots)

-First and last knots are duplicated k times.
-E.g (0,0,0,1,2,2,2)
-Curve pass through the first and last control points

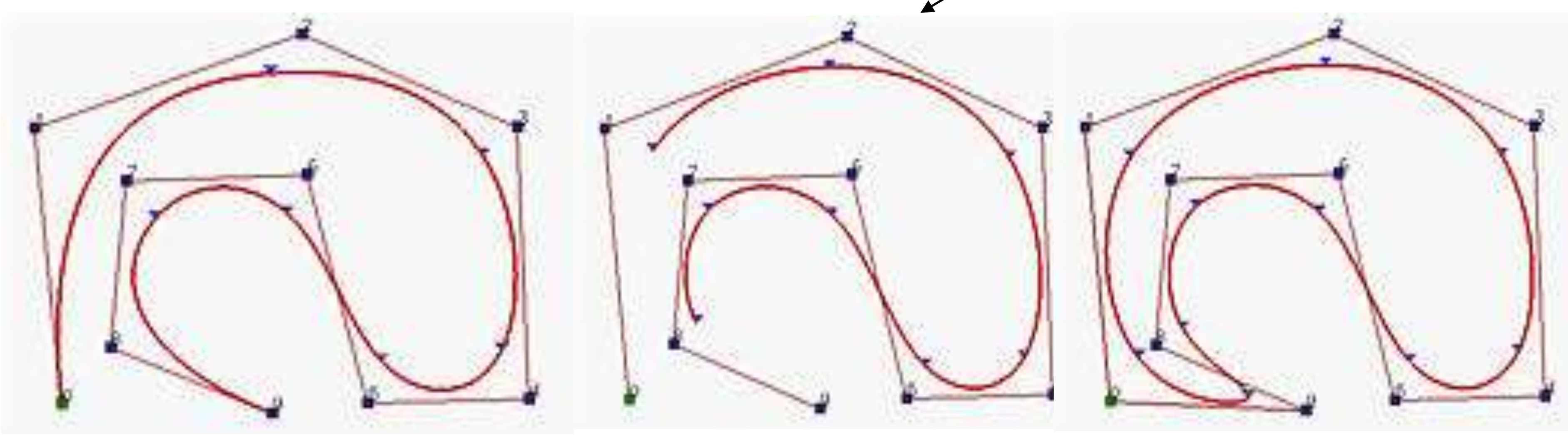
Periodic knots (non-open knots)

-First and last knots are not duplicated – same contribution.
-E.g (0, 1, 2, 3)
-Curve doesn’t pass through end points.
- used to generate closed curves (when first = last control points)

Type of B-Spline Uniform knot vector

Non-periodic knots (open knots)

Periodic knots (non-open knots)



(Closed knots)

Non-periodic (open) uniform B-Spline

- The knot spacing is evenly spaced except at the ends where knot values are repeated *k* times.
- E.g $P(u) = \sum_{i=0}^{n} N_{i,k}(u)p_i \qquad (u_0 \le u \le u_m)$
- Degree = k-1, number of control points = n + 1
- Number of knots = m + 1 @ n+ k + 1

→for degree = 1 and number of control points = 4 →(k = 2, n = 3)

→Number of knots = n + k + 1 = 6

→ Range = 0 to n+k

non periodic uniform knot vector (0,0,1,2,3, 3)

* Knot value between 0 and 3 are equally spaced → uniform

Questions

- For curve degree = 3, number of control points = 5
- For curve degree = 1, number of control points = 5
- k = ? , n = ? , Range = ?

  Knot vector = ?

Non-periodic (open) uniform B-Spline

- Example
- For curve degree = 3, number of control points = 5
- → k = 4, n = 4
- → number of knots = n+k+1 = 9
- → non periodic knots vector = (0,0,0,0,1,2,2,2,2)
- For curve degree = 1, number of control points = 5
- → k = 2, n = 4
- → number of knots = n + k + 1 = 7
- → non periodic uniform knots vector = (0, 0, 1, 2, 3, 4, 4)

Non-periodic (open) uniform B-Spline

- For any value of parameters k and n, non periodic knots are determined from

$$u_i = \begin{cases} 0 & 0 \le i < k \\ i - k + 1 & k \le i \le n \\ n - k + 2 & n < i \le n+k \end{cases} \qquad (1.3)$$

e.g k=2, n = 3

$$u_i = \begin{cases} 0 & 0 \le i < 2 \\ i - 2 + 1 & 2 \le i \le 3 \\ 3 - 2 + 2 & 3 < i \le 5 \end{cases}$$

u = (0, 0, 1, 2, 3, 3)

B-Spline basis function

$$N_{i,k}(u) = (u - u_i)\frac{N_{i,k-1}(u)}{u_{i+k-1} - u_i} + (u_{i+k} - u)\frac{N_{i+1,k-1}(u)}{u_{i+k} - u_{i+1}} \qquad (1.1)$$

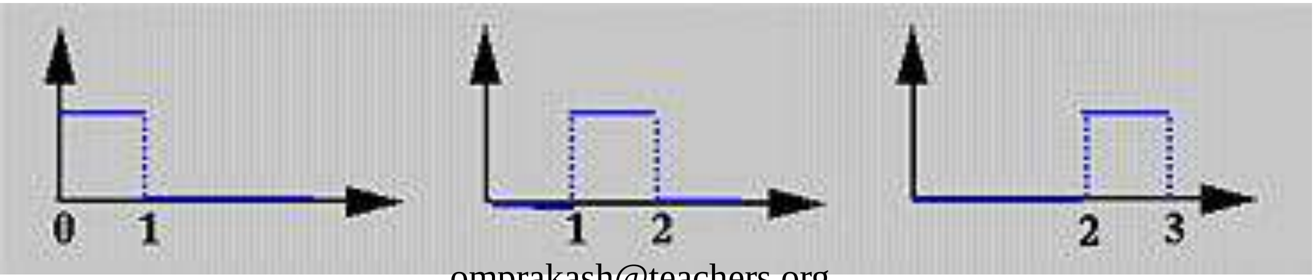
$$N_{i,1} = \begin{cases} 1 & u_i \le u \le u_{i+1} \\ 0 & \text{Otherwise} \end{cases} \qquad (1.2)$$

→In equation (1.1), the denominators can have a value of zero, 0/0 is presumed to be zero.

→If the degree is zero basis function $N_{i,1}(u)$ is 1 if u is in the i-th knot span $[u_i, u_{i+1})$.

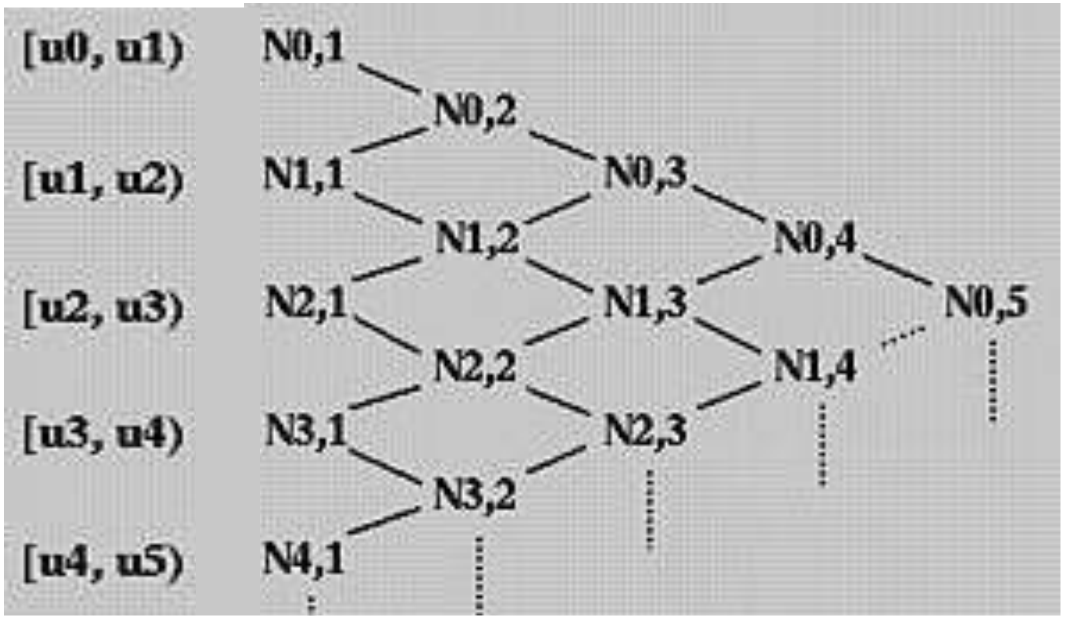
B-Spline basis function

- For example, if we have four knots $u_0 = 0$, $u_1 = 1$, $u_2 = 2$ and $u_3 = 3$, knot spans 0, 1 and 2 are [0,1), [1,2), [2,3)
- the basis functions of degree 0 are $N_{0,1}(u) = 1$ on [0,1) and 0 elsewhere, $N_{1,1}(u) = 1$ on [1,2) and 0 elsewhere, and $N_{2,1}(u) = 1$ on [2,3) and 0 elsewhere.
- This is shown below



B-Spline basis function

- To understand the way of computing $N_{i,k}(u)$ for k greater than 0, we use the triangular computation scheme



Non-periodic (open) uniform B-Spline

Example

- Find the knot values of a non periodic uniform B-Spline which has degree = 2 and 3 control points. Then, find the equation of B-Spline curve in polynomial form.

Non-periodic (open) uniform B-Spline

Answer

- Degree = k-1 = 2 → k=3
- Control points = n + 1 = 3 → n=2
- Number of knot = n + k + 1 = 6
- Knot values → 0,0,0,1,1,1

Non-periodic (open) uniform B-Spline

Answer(cont)

- To obtain the polynomial equation,
  $P(u) = \sum_{i=0}^{n} N_{i,k}(u)p_i$
- $= \sum_{i=0}^{2} N_{i,3}(u)p_i$
- $= N_{0,3}(u)p_0 + N_{1,3}(u)p_1 + N_{2,3}(u)p_2$

- firstly, find the $N_{i,k}(u)$ using the knot value that shown above, start from k =1 to k=3

Non-periodic (open) uniform B-Spline

Answer (cont)

- For k = 1, find $N_{i,1}(u)$ – use equation (1.2):
  - $N_{0,1}(u) = \begin{cases} 1 & u_0 \leq u \leq u_1 \quad ; \ (u=0) \\ 0 & \text{otherwise} \end{cases}$
  - $N_{1,1}(u) = \begin{cases} 1 & u_1 \leq u \leq u_2 \quad ; \ (u=0) \\ 0 & \text{otherwise} \end{cases}$
  - $N_{2,1}(u) = \begin{cases} 1 & u_2 \leq u \leq u_3 \quad ; \ (0 \leq u \leq 1) \\ 0 & \text{otherwise} \end{cases}$
  - $N_{3,1}(u) = \begin{cases} 1 & u_3 \leq u \leq u_4 \quad ; \ (u=1) \\ 0 & \text{otherwise} \end{cases}$
  - $N_{4,1}(u) = \begin{cases} 1 & u_4 \leq u \leq u_5 \quad ; \ (u=1) \\ 0 & \text{otherwise} \end{cases}$

Non-periodic (open) uniform B-Spline

Answer (cont)

- For k = 2, find $N_{i,2}(u)$ – use equation (1.1):

$$N_{i,k}(u) = (u - u_i)\frac{N_{i,k-1}(u)}{u_{i+k-1} - u_i} + (u_{i+k} - u)\frac{N_{i+1,k-1}(u)}{u_{i+k} - u_{i+1}}$$

- $N_{0,2}(u) = \dfrac{u - u_0}{u_1 - u_0} N_{0,1} + \dfrac{u_2 - u}{u_2 - u_1} N_{1,1}$ $\quad (u_0 = u_1 = u_2 = 0)$
- $= \dfrac{u - 0}{0 - 0} N_{0,1} + \dfrac{0 - u}{0 - 0} N_{1,1} = 0$
- $N_{1,2}(u) = \dfrac{u - u_1}{u_2 - u_1} N_{1,1} + \dfrac{u_3 - u}{u_3 - u_2} N_{2,1}$ $\quad (u_1 = u_2 = 0, u_3 = 1)$
- $= \dfrac{u - 0}{0 - 0} N_{1,1} + \dfrac{1 - u}{1 - 0} N_{2,1} = 1 - u$

Non-periodic (open) uniform B-Spline

Answer (cont)

- $N_{2,2}(u) = \dfrac{u - u_2}{u_3 - u_2} N_{2,1} + \dfrac{u_4 - u}{u_4 - u_3} N_{3,1}$ $(u_2 = 0, u_3 = u_4 = 1)$
- $= \dfrac{u - 0}{1 - 0} N_{2,1} + \dfrac{1 - u}{1 - 1} N_{3,1} = u$
- $N_{3,2}(u) = \dfrac{u - u_3}{u_4 - u_3} N_{3,1} + \dfrac{u_5 - u}{u_5 - u_4} N_{4,1}$ $(u_3 = u_4 = u_5 = 1)$
- $= \dfrac{u - 1}{1 - 1} N_{3,1} + \dfrac{1 - u}{1 - 1} N_{4,1} = 0$

Non-periodic (open) uniform B-Spline

Answer (cont)

For k = 2

$N_{0,2}(u) = 0$

$N_{1,2}(u) = 1 - u$

$N_{2,2}(u) = u$

$N_{3,2}(u) = 0$

Non-periodic (open) uniform B-Spline

Answer (cont)

- For k = 3, find $N_{i,3}(u)$ – use equation (1.1):

$$N_{i,k}(u) = (u - u_i)\frac{N_{i,k-1}(u)}{u_{i+k-1} - u_i} + (u_{i+k} - u)\frac{N_{i+1,k-1}(u)}{u_{i+k} - u_{i+1}}$$

- $N_{0,3}(u) = \frac{u - u_0}{u_2 - u_0} N_{0,2} + \frac{u_3 - u}{u_3 - u_1} N_{1,2}$ $(u_0 = u_1 = u_2 = 0, u_3 = 1)$
- $= \frac{u - 0}{0 - 0} N_{0,2} + \frac{1 - u}{1 - 0} N_{1,2} = (1-u)(1-u) = (1-u)^2$
- $N_{1,3}(u) = \frac{u - u_1}{u_3 - u_1} N_{1,2} + \frac{u_4 - u}{u_4 - u_2} N_{2,2}$ $(u_1 = u_2 = 0, u_3 = u_4 = 1)$
- $= \frac{u - 0}{1 - 0} N_{1,2} + \frac{1 - u}{1 - 0} N_{2,2} = u(1 - u) + (1-u)u = 2u(1-u)$

Non-periodic (open) uniform B-Spline

Answer (cont)

- $N_{2,3}(u) = \dfrac{u - u_2}{u_4 - u_2} N_{2,2} + \dfrac{u_5 - u}{u_5 - u_3} N_{3,2}$ $(u_2 = 0,\ u_3 = u_4 = u_5 = 1)$
- $= \dfrac{u - 0}{1 - 0} N_{2,2} + \dfrac{1 - u}{1 - 1} N_{3,2} = u^2$

$N_{0,3}(u) = (1- u)^2,$ $N_{1,3}(u) = 2u(1-u),$ $N_{2,3}(u) = u^2$

The polynomial equation, $P(u) = \sum_{i=0}^{n} N_{i,k}(u) p_i$

- $P(u) = N_{0,3}(u)p_0 + N_{1,3}(u)p_1 + N_{2,3}(u)p_2$
- **$= (1- u)^2 p_0 + 2u(1-u)\, p_1 + u^2 p_2$ $(0 <= u <= 1)$**

Non-periodic (open) uniform B-Spline

- Exercise
- Find the polynomial equation for curve with degree = 1 and number of control points = 4

Non-periodic (open) uniform B-Spline

- Answer
- k = 2 , n = 3 → number of knots = 6
- Knot vector = (0, 0, 1, 2, 3, 3)
- For k = 1, find $N_{i,1}(u)$ – use equation (1.2):
  - $N_{0,1}(u) = 1 \quad u_0 \le u \le u_1$; (u=0)
  - $N_{1,1}(u) = 1 \quad u_1 \le u \le u_2$; $(0 \le u \le 1)$
    $N_{2,1}(u) = 1 \quad u_2 \le u \le u_3$; $(1 \le u \le 2)$
  - $N_{3,1}(u) = 1 \quad u_3 \le u \le u_4$; $(2 \le u \le 3)$
    $N_{4,1}(u) = 1 \quad u_4 \le u \le u_5$; (u=3)
  -

Non-periodic (open) uniform B-Spline

Answer (cont)

- For k = 2, find $N_{i,2}(u)$ – use equation (1.1):

$$N_{i,k}(u) = (u - u_i)\frac{N_{i,k-1}(u)}{u_{i+k-1} - u_i} + (u_{i+k} - u)\frac{N_{i+1,k-1}(u)}{u_{i+k} - u_{i+1}}$$

- $N_{0,2}(u) = \dfrac{u - u_0}{u_1 - u_0} N_{0,1} + \dfrac{u_2 - u}{u_2 - u_1} N_{1,1}$ $\quad (u_0 = u_1 = 0,\ u_2 = 1)$
- $= \dfrac{u - 0}{0 - 0} N_{0,1} + \dfrac{1 - u}{1 - 0} N_{1,1}$
- $= 1 - u \quad (0 \le u \le 1)$

omprakash@teachers.org

Non-periodic (open) uniform B-Spline

Answer (cont)

- For k = 2, find $N_{i,2}(u)$ – use equation (1.1):

$$N_{i,k}(u) = (u - u_i)\frac{N_{i,k-1}(u)}{u_{i+k-1} - u_i} + (u_{i+k} - u)\frac{N_{i+1,k-1}(u)}{u_{i+k} - u_{i+1}}$$

- $N_{1,2}(u) = \frac{u - u_1}{u_2 - u_1} N_{1,1} + \frac{u_3 - u}{u_3 - u_2} N_{2,1}$ $(u_1 = 0,\ u_2 = 1,\ u_3 = 2)$
- $= \frac{u - 0}{1 - 0} N_{1,1} + \frac{2 - u}{2 - 1} N_{2,1}$
- $N_{1,2}(u) = u$ $(0 \le u \le 1)$
- $N_{1,2}(u) = 2 - u$ $(1 \le u \le 2)$

Non-periodic (open) uniform B-Spline

Answer (cont)

- $N_{2,2}(u) = \dfrac{u - u_2}{u_3 - u_2} N_{2,1} + \dfrac{u_4 - u}{u_4 - u_3} N_{3,1}$ $(u_2 = 1, u_3 = 2, u_4 = 3)$
- $= \dfrac{u - 1}{2 - 1} N_{2,1} + \dfrac{3 - u}{3 - 2} N_{3,1} =$
- $N_{2,2}(u) = u - 1 \quad (1 \le u \le 2)$
- $N_{2,2}(u) = 3 - u \quad (2 \le u \le 3)$

Non-periodic (open) uniform B-Spline

Answer (cont)

- $N_{3,2}(u) = \dfrac{u - u_3}{u_4 - u_3} N_{3,1} + \dfrac{u_5 - u}{u_5 - u_4} N_{4,1}$ $(u_3 = 2, u_4 = 3, u_5 = 3)$
- $= \dfrac{u - 2}{3 - 2} N_{3,1} + \dfrac{3 - u}{3 - 3} N_{4,1} =$
- $= u - 2 \quad (2 \leq u \leq 3)$

Non-periodic (open) uniform B-Spline

Answer (cont)

- The polynomial equation $P(u) = \sum N_{i,k}(u)p_i$
- $P(u) = N_{0,2}(u)p_0 + N_{1,2}(u)p_1 + N_{2,2}(u)p_2 + N_{3,2}(u)p_3$

- $P(u) = (1 – u)\ p_0 + u\ p_1$ $(0 \le u \le 1)$
- $P(u) = (2 – u)\ p_1 + (u – 1)\ p_2$ $(1 \le u \le 2)$
- $P(u) = (3 – u)\ p_2 + (u - 2)\ p_3$ $(2 \le u \le 3)$

Periodic uniform knot

- Periodic knots are determined from
  - U_i ; $(0 \le i \le n+k)$
- Example
  - For curve with degree = 3 and number of control points = 4 (cubic B-spline)
  - (k = 4, n = 3) → number of knots = n+k+1 =8
  - (0, 1, 2, 3, 4, 5, 6, 7)

Periodic uniform knot

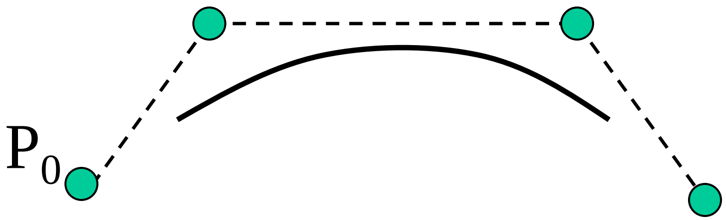
- Normalize u (0<= u <= 1)
- $N_{0,4}(u) = 1/6\ (1-u)^3$
- $N_{1,4}(u) = 1/6\ (3u^3 - 6u^2 + 4)$
- $N_{2,4}(u) = 1/6\ (-3u^3 + 3u^2 + 3u + 1)$
- $N_{3,4}(u) = 1/6\ u^3$

- $P(u) = N_{0,4}(u)p_0 + N_{1,4}(u)p_1 + N_{2,4}(u)p_2 + N_{3,4}(u)p_3$

Periodic uniform knot

- In matrix form
- $P(u) = [u^3, u^2, u, 1].M_n.\begin{pmatrix} P_0 \\ P_1 \\ P_2 \\ P_3 \end{pmatrix}$
- $M_n = 1/6 \begin{pmatrix} -1 & 3 & -3 & 1 \\ 3 & -6 & 3 & 0 \\ -3 & 0 & 3 & 0 \\ 1 & 4 & 1 & 0 \end{pmatrix}$

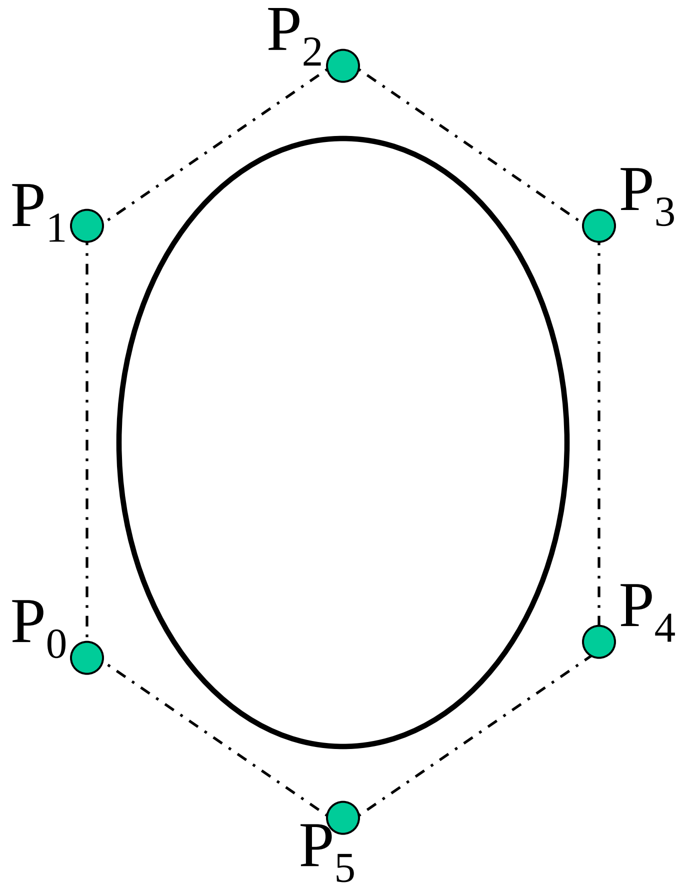
Periodic uniform knot



Closed periodic

Example

k = 4, n = 5



Closed periodic

Equation 1.0 change to

- $N_{i,k}(u) = N_{0,k}((u-i)\mathrm{mod}(n+1))$

$\rightarrow P(u) = \sum_{i=0}^{n} N_{0,k}((u-i)\mathrm{mod}(n+1))p_i$

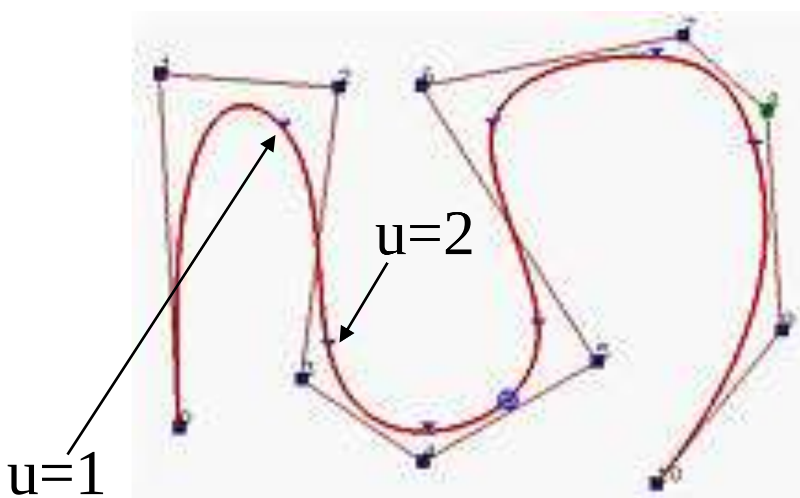
$0 <= u <= n+1$

Question 1

Construct the B-Spline curve of degree/order 3 with 4 polygon vertices A(1,1), B(2,3), C(4,3) and D(6,2). Using Non-Periodic Knot and Periodic Knot.

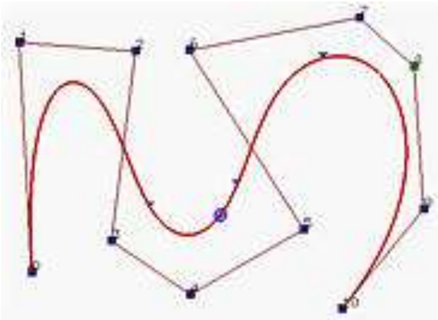
Properties of B-Spline

1. The m degree B-Spline function are piecewise polynomials of degree m → have C^{m-1} continuity. →e.g B-Spline degree 3 have C^2 continuity.

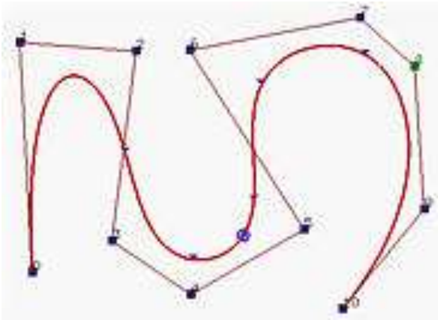


Properties of B-Spline

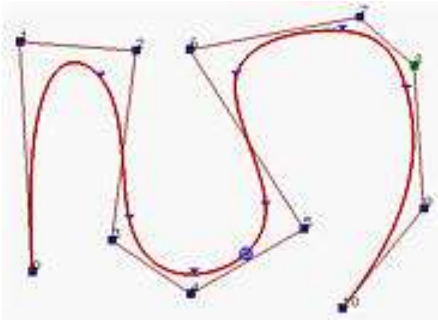
In general, the lower the degree, the closer a B-spline curve follows its control polyline.



Degree = 7



Degree = 5



Degree = 3

Properties of B-Spline

Equality $m = n + k$ must be satisfied

Number of knots = m + 1

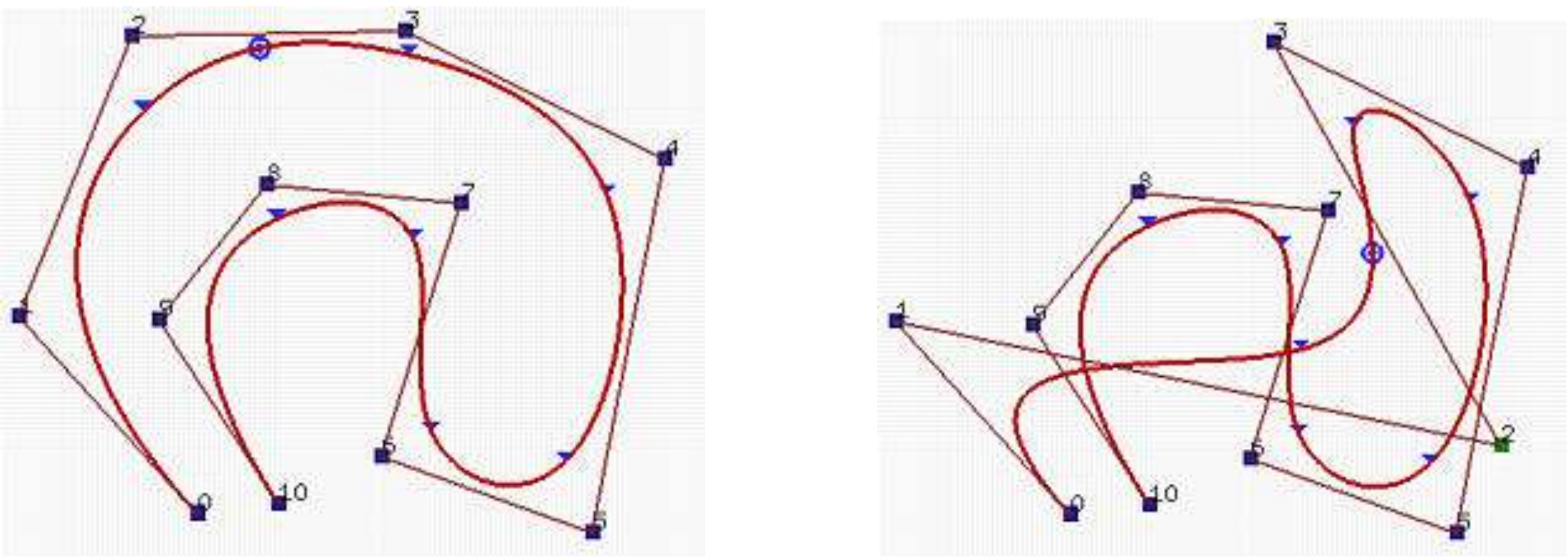
k cannot exceed the number of control points, n+ 1

Properties of B-Spline

2. Each curve segment is affected by k control points as shown by past examples.→ e.g k = 3, $P(u) = N_{i-1,k}\, p_{i-1} + N_{i,k}\, p_i + N_{i+1,k}\, p_{i+1}$

Properties of B-Spline

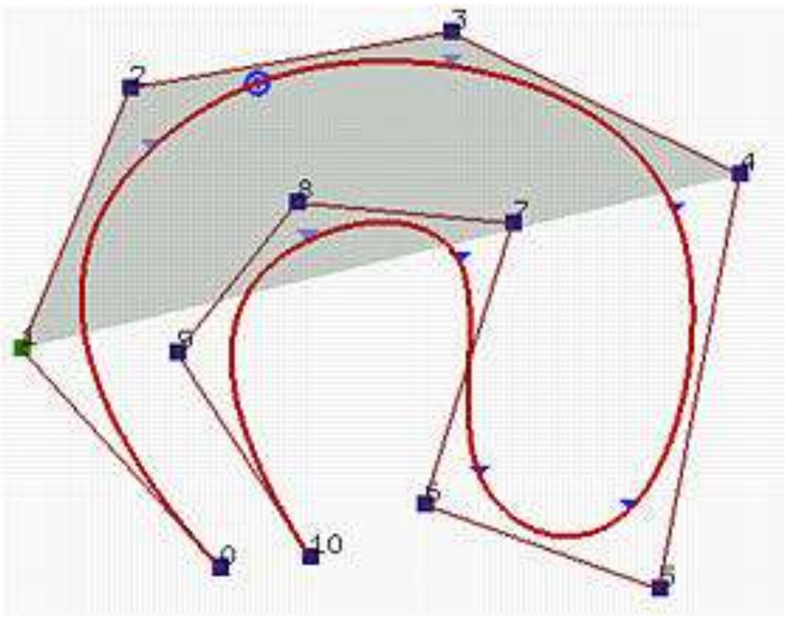
Local Modification Scheme: changing the position of control point P_i only affects the curve $C(u)$ on interval $[u_i, u_{i+k})$.



Modify control point P_2

Properties of B-Spline

3. Strong Convex Hull Property: A B-spline curve is contained in the convex hull of its control polyline. More specifically, if u is in knot span $[u_i, u_{i+1})$, then $C(u)$ is in the convex hull of control points P_{i-p}, P_{i-p+1}, ..., P_i.



Degree = 3, k = 4
Convex hull based on 4 control points

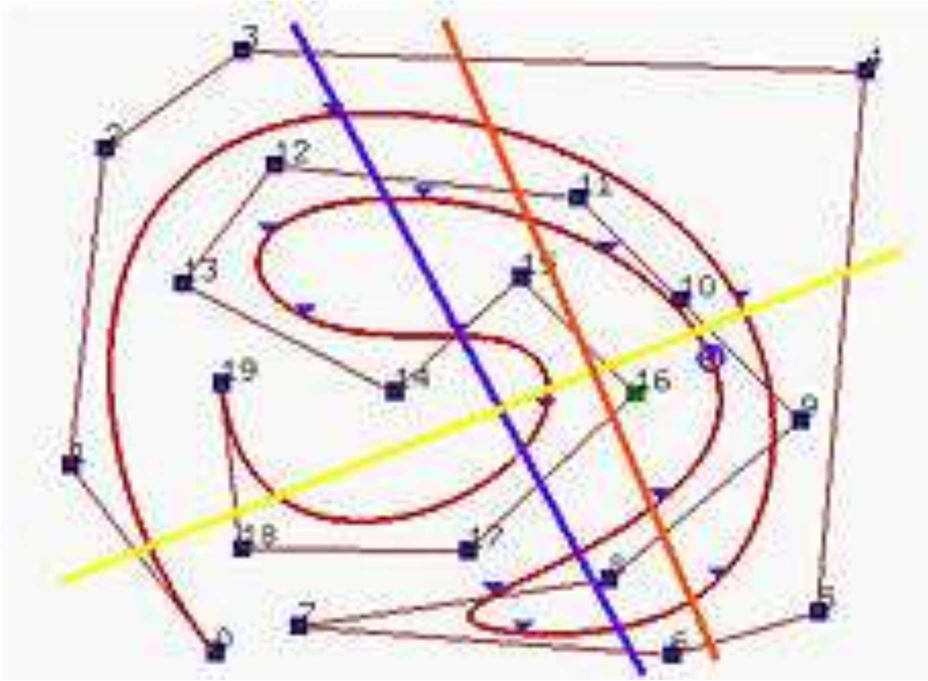
Properties of B-Spline

4. Non-periodic B-spline curve C(*u*) passes through the two end control points P_0 and P_n.
5. Each B-spline function Nk,m(t) is nonnegative for every t, and the family of such functions sums to unity, that is $\sum_{i=0}^{n} N_{i,k}(u) = 1$
6. Affine Invariance

   to transform a B-Spline curve, we simply transform each control points.
7. Bézier Curves Are Special Cases of B-spline Curves

Properties of B-Spline

8. Variation Diminishing : A B-Spline curve does not pass through any line more times than does its control polyline



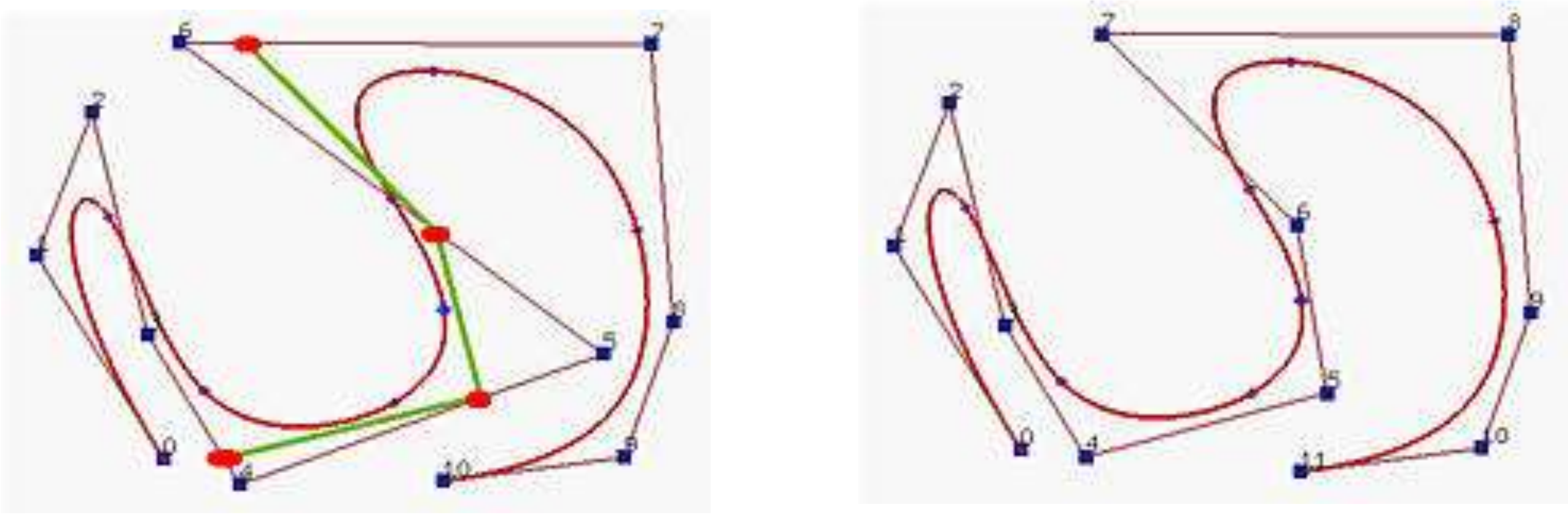
Knot Insertion : B-Spline

- *knot insertion* is adding a new knot into the existing knot vector without changing the shape of the curve.
- new knot may be equal to an existing knot → the multiplicity of that knot is increased by one
- Since, number of knots = k + n + 1
- If the number of knots is increased by 1→ either degree or number of control points must also be increased by 1.
- Maintain the curve shape →maintain degree →change the number of control points.

omprakash@teachers.org

Knot Insertion : B-Spline

- So, inserting a new knot causes a new control point to be added. In fact, some existing control points are removed and replaced with new ones by corner cutting



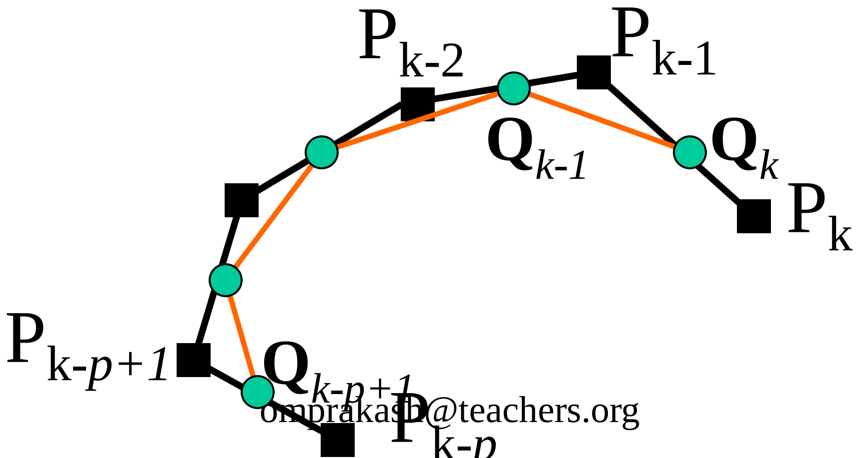
Insert knot u = 0.5

Single knot insertion : B-Spline

- Given n+1 control points – P_0, P_1, .. P_n
- Knot vector, $U= (u_0, u_1,\ldots u_m)$
- Degree = p, order, k = p+1
- Insert a new knot t into knot vector without changing the shape.
- → find the knot span that contains the new knot. Let say $[u_k, u_{k+1})$

Single knot insertion : B-Spline

- This insertion will affected to k (degree + 1) control points (refer to B-Spline properties) → P_k, P_{k-1}, P_{k-1},…P_{k-p}
- Find p new control points $\mathbf{Q}_k$ on leg $\mathbf{P}_{k-1}\mathbf{P}_k$, $\mathbf{Q}_{k-1}$ on leg $\mathbf{P}_{k-2}\mathbf{P}_{k-1}$, ..., and $\mathbf{Q}_{k-p+1}$ on leg $\mathbf{P}_{k-p}\mathbf{P}_{k-p+1}$ such that the old polyline between $\mathbf{P}_{k-p}$ and $\mathbf{P}_k$ (in black below) is replaced by $\mathbf{P}_{k-p}\mathbf{Q}_{k-p+1}...\mathbf{Q}_k\mathbf{P}_k$ (in orange below)



Single knot insertion : B-Spline

- All other control points are not change
- The formula for computing the new control point $\mathbf{Q}_i$ on leg $\mathbf{P}_{i-1}\mathbf{P}_i$ is the following
  - $\mathbf{Q}_i = (1-a_i)\mathbf{P}_{i-1} + a_i\mathbf{P}_i$
  - $a_i = \frac{t - u_i}{u_{i+p} - u_i}$ $\quad k-p+1 <= i <= k$

Single knot insertion : B-Spline

- Example
- Suppose we have a B-spline curve of degree 3 with a knot vector as follows:

u_0 to u_3	u_4	u_5	u_6	u_7	u_8 to u_{11}
0	0.2	0.4	0.6	0.8	1

Insert a new knot t = 0.5 , find new control points and new knot vector?

Single knot insertion : B-Spline

Solution:

- $t = 0.5$ lies in knot span $[u_5, u_6)$
- the affected control points are $\mathbf{P}_5$, $\mathbf{P}_4$, $\mathbf{P}_3$ and $\mathbf{P}_2$
- find the 3 new control points Q_5, Q_4, Q_3
- we need to compute a_5, a_4 and a_3 as follows
  - $a_5 = \frac{t - u_5}{u_8 - u_5} = \frac{0.5 - 0.4}{1 - 0.4} = 1/6$
  - $a_4 = \frac{t - u_4}{u_7 - u_4} = \frac{0.5 - 0.2}{0.8 - 0.2} = 1/2$
  - $a_3 = \frac{t - u_3}{u_6 - u_3} = \frac{0.5 - 0}{0.6 - 0} = 5/6$

Single knot insertion : B-Spline

- Solution (cont)
- The three new control points are
- $\mathbf{Q}_5 = (1-a_5)\mathbf{P}_4 + a_5\mathbf{P}_5 = (1-1/6)\mathbf{P}_4 + 1/6\mathbf{P}_5$
- $\mathbf{Q}_4 = (1-a_4)\mathbf{P}_3 + a_4\mathbf{P}_4 = (1-1/6)\mathbf{P}_3 + 1/6\mathbf{P}_4$
- $\mathbf{Q}_3 = (1-a_3)\mathbf{P}_2 + a_3\mathbf{P}_3 = (1-5/6)\mathbf{P}_2 + 5/6\mathbf{P}_3$

Single knot insertion : B-Spline

- Solution (cont)
- The new control points are $\mathbf{P}_0$, $\mathbf{P}_1$, $\mathbf{P}_2$, $\mathbf{Q}_3$, $\mathbf{Q}_4$, $\mathbf{Q}_5$, $\mathbf{P}_5$, $\mathbf{P}_6$, $\mathbf{P}_7$
- the new knot vector is

u_0 to u_3	u_4	u_5	u_6	u_7	u_8	u_9 to u_{12}
0	0.2	0.4	0.5	0.6	0.8	1

RATIONAL SPLINES

A rational function is simply the ratio of two polynomials. Thus, a **rational spline** is the ratio of two spline functions. For example, a rational B-spline curve can be described with the position vector:

$$\mathbf{P}(u) = \frac{\sum_{k=0}^{n} \omega_k \mathbf{p}_k B_{k,d}(u)}{\sum_{k=0}^{n} \omega_k B_{k,d}(u)} \qquad (10\text{-}69)$$

where the $\mathbf{p}_k$ are a set of $n + 1$ control-point positions. Parameters ω_k are weight factors for the control points. The greater the value of a particular ω_k, the closer the curve is pulled toward the control point $\mathbf{p}_k$ weighted by that parameter. When all weight factors are set to the value 1, we have the standard B-spline curve since the denominator in Eq. 10-69 is 1 (the sum of the blending functions).

To plot conic sections with NURBs, we use a quadratic spline function (d = 3) and three control points. We can do this with a B-spline function defined with the open knot vector:

$$\{0, 0, 0, 1, 1, 1\}$$

which is the same as a quadratic Bézier spline. We then set the weighting functions to the following values:

$$\omega_0 = \omega_2 = 1$$
$$\omega_1 = \frac{r}{1-r}, \quad 0 \leq r < 1 \qquad (10\text{-}70)$$

cp = 3
Degree = 2
k = 3
n =2

and the rational B-spline representation is

$$\mathbf{P}(u) = \frac{\mathbf{p}_0 B_{0,3}(u) + [r/(1-r)]\mathbf{p}_1 B_{1,3}(u) + \mathbf{p}_2 B_{2,3}(u)}{B_{0,3}(u) + [r/(1-r)] B_{1,3}(u) + B_{2,3}(u)} \qquad (10\text{-}71)$$

We then obtain the various conics (Fig. 10-50) with the following values for parameter r:

$r > 1/2, \quad \omega_1 > 1$ (hyperbola section)

$r = 1/2, \quad \omega_1 = 1$ (parabola section)

$r < 1/2, \quad \omega_1 < 1$ (ellipse section)

$r = 0, \quad \omega_1 = 0$ (straight-line segment)

Example: A full circle can be obtained by using seven control points:

$$\{P_0, P_1, P_2, P_3, P_4, P_5, P_6\}$$

Solution :

Degree = 6

Degree = k-1 ; k = 7

Control Points = n+1 ; 7= n +1 ; n=6

Range = n+k = 13 ;

Knot Value = n+k+1 = 6+7+1 = 14

Weight = 7 (Control Point = Weight)

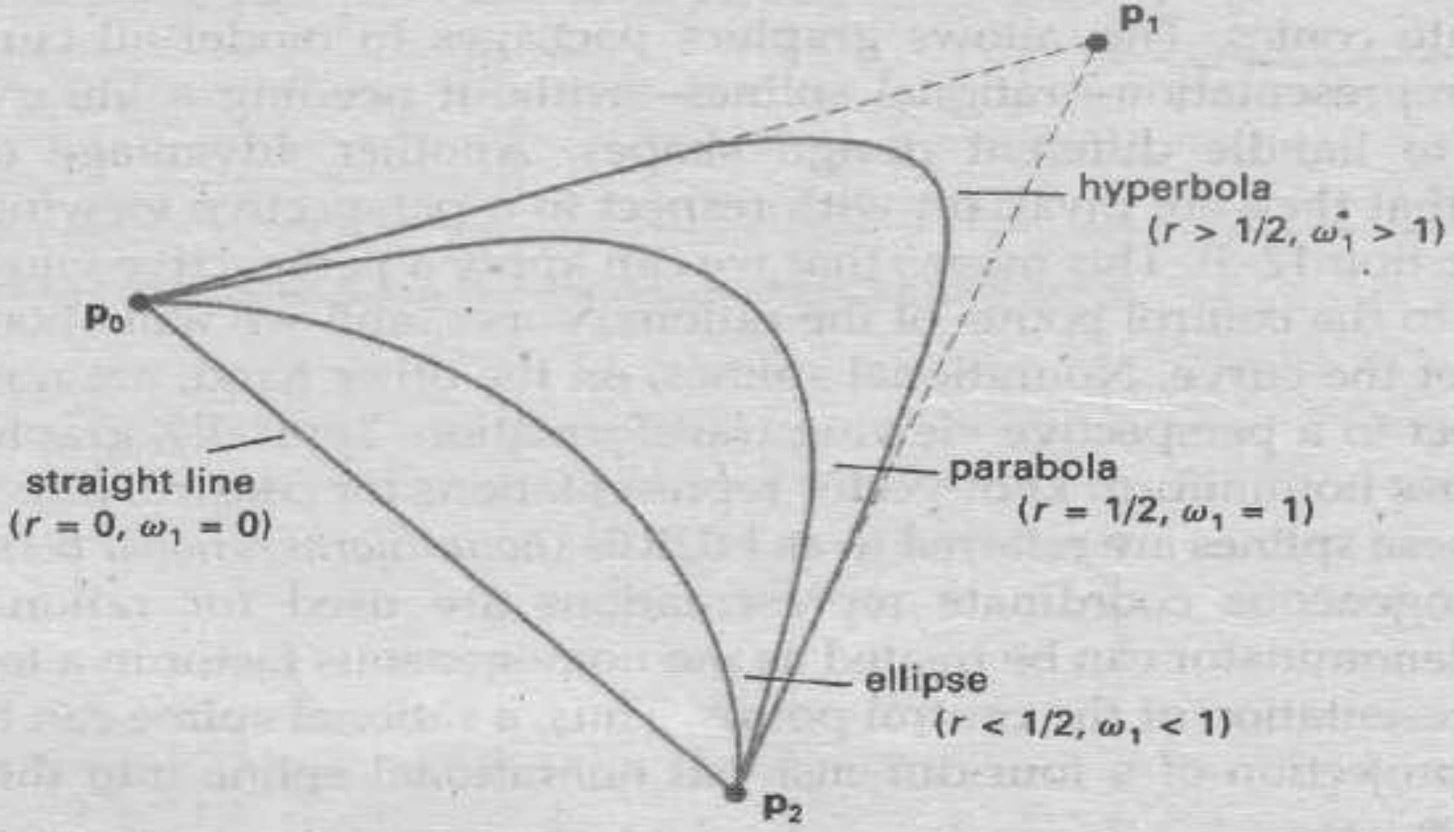


Figure 10-50

Conic sections generated with various values of the rational-spline weighting factor ω_1.

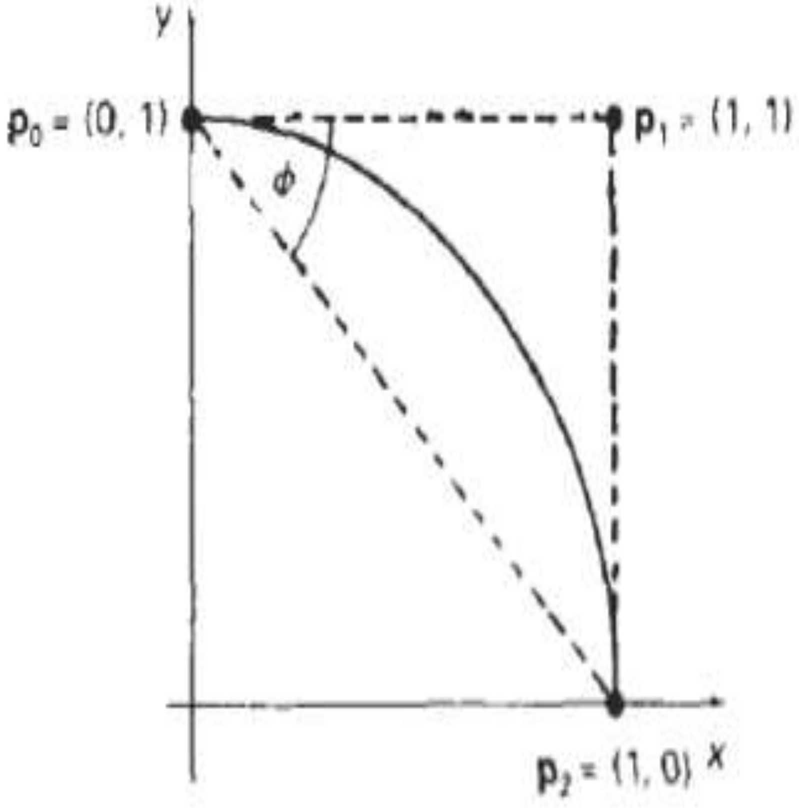


Figure 10-51
A circular arc in the first quadrant of the *xy* plane.

$$p_0 = (0, 1), \quad p_1 = (1, 1), \quad p_2 = (1, 0)$$

Question :

Calculate the k, n, total number of knots, Knot Values/Vectors, range and Weight on followings :

1. Control Point = 5
   Degree = 4

2. Control Point = 6

3. Degree = 3

Beta-Splines:

Subdivision Methods

Drawing curves using forward differences

DO YOU KNOW

1-99 ………………………….. No A,B,C
100………………………….. Only D

1-999 ………………………….. No A,B,C
1000………………………….. Only A

1-999,999,999 ………………… No B,C
Billion……………………….. Only B

There is no entry of C in Table (CRORE)