# COMPUTER AIDED DESIGN
## (BME-42)

## Unit-III: Planar Curves

**(1 Lecture)**

- Curves representation
- Interpolation vs approximation
- Classical representation of curves
- Parametric analytic curves-lines, circles, ellipses, parabolas and hyperbolas

# Lecture 19

## Topics Covered

**Plane Curve**

**Curve Representation**

**Interpolation & Approximation**

**Classical Representation of Curves**

Nonparametric Curves

Parametric Curves

Lines, Circles, Ellipse,
Parabola and Hyperbola

*Prepared By*


Prof. S. K. SRIVASTAVA

MED, MMMUT, Gorakhpur (UP)
sksme@mmmut.ac.in

# PLANE CURVES

- A curve is an integral part of any design and an engineer needs to draw one or other type of curves or curved surfaces applicable to many engineering components used in automotive, aerospace and hydrospace industries.

- Different types of shape constraints (e.g., *continuity* and/or *curvature*) are imposed to accomplish specific shapes of the curve or curved surfaces.

- A large variety of techniques are available for drawing and sketching the curves manually with tools like pencils, brushes, knives, French curves, compasses, splines, templates, etc.

- Each tool is used for the specific work. None of the single tool is sufficient to draw all types of curves.

- When a curve is two-dimensional, it lies entirely in a plane known as ***planar*** curve.

- However, three-dimensional curve lies in space called ***space*** curve.

# PLANE CURVES…

- In general, engineering applications require smooth curves. A smooth curve can be generated by *reducing the spacing between the data points.*
- The data points may be *equally spaced* or *non-evenly spaced* (Figure).
- The connection of evenly spaced points by small line segments results into poor representation of a curve.
- The advantage of non-evenly spaced data points can be visualized for generating the smooth curve as shown in Figure.
- *Increasing the density of data points in the region where radius of the curvature is small improves the quality, as regards to the smoothness, of the curve*.


**(a) Evenly Spaced data points (b) Non-evenly spaced data points**

# CURVE REPRESENTATION

There are **two** techniques for the curve representation.

## Analytic Curves

Analytic curves can be represented by the analytical (mathematical) equations such as lines, circles and conics. This type of curve representation has the following advantages:

- Precise and easy evaluation of the intermediate points
- Mathematical representation of curve is computer friendly, i.e., compact storage of curve
- Curve properties such as *slope* and *radius of curvature* can be easily evaluated
- Drawing of curves is easy from the storage data
- *Alteration / manipulation* of curve is easy to meet the modified design criteria

## Synthetic Curves

Unfortunately, it is not possible to represent all types of curves required in engineering applications analytically; therefore, the method based on the data points (**synthetic curves**) is very useful in designing the objects with curved shapes such as ship hull, car body, aerofoil section, automobile components, etc.

- Synthetic curves such as Bezier curves and splines are described by a set of data points known as control points.
- Parametric polynomials usually fit the control points.
- Synthetic curves provide greater flexibility to the designer just by changing the positions of the control points.
- Moreover, it is possible to achieve a **local control** and **global control** of the shape of the curve.

## Synthetic Curves

The data (control) point representation of curve suffers from the following disadvantages:

- Slope of the curve is obtained using numerical differentiation, a well-known inaccurate procedure.
- A good quality circle requires a minimum of 32 points on its circumference; therefore, a huge storage is required as compared to the analytical representation of circle in which centre and radius is sufficient to represent the circle.
- Intermediate points are obtained using the interpolation techniques. The resulting intermediate points do not actually lie on the curve.
- It is not possible to calculate the exact property of the curve because exact shape of the curve is not known.
- Difficult to handle the transformations of curve due to the large number of data points.

## Interpolation

- The *interpolation* is a technique by which a curve, represented with known set of data points, can be defined analytically.
- The data points may be obtained through the experimental measurements or from some known function.
- When curve passes through all the data points, it is said to ***fit*** the data.
- 'Piecewise polynomial approximation' technique of ***curve fitting*** is used to determine the coefficient of polynomials of some degree.
- The curve shape between the data points depends upon the *degree of polynomial* and the *associated boundary conditions*.

## Approximation

- If data points (control points) are only approximation to some true values (e.g., measurement points, etc.) then the curve does not necessarily pass through the data points rather than it *approximates* or *fairs* the data points.

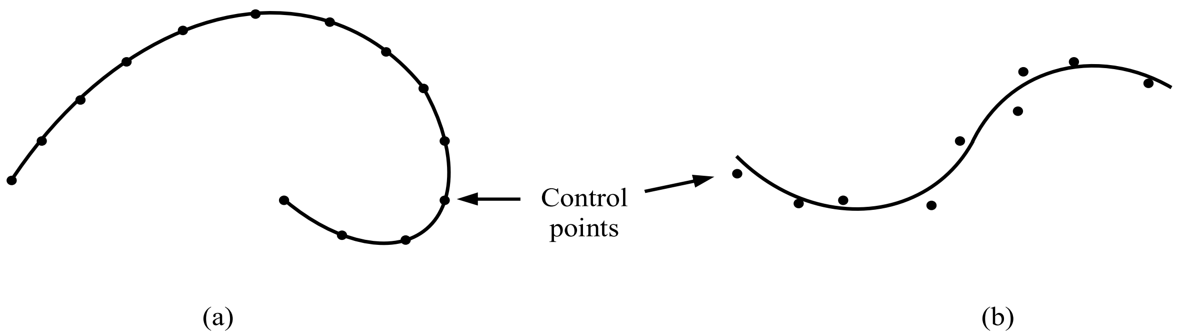
- The curve depicts the trend of data points.

- *Least square approximation* is a common *curve fairing* technique, which produces the curve of the form $y = f(x)$ , which minimizes the sum of $y$ squared deviations between the data and the derived curve.

- Depending upon the information about the phenomenon that produces the data points, the curve $y = f(x)$ may be

# INTERPOLATION AND APPROXIMATION…

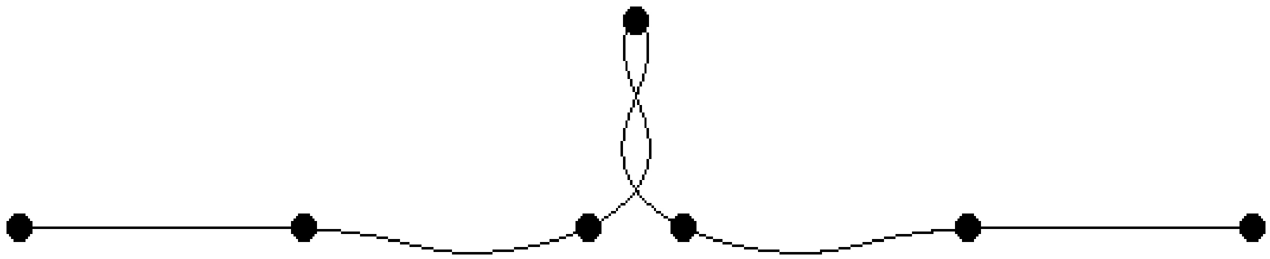
- Power functions $y = a.x^b$
- Exponential functions $y = a.e^{bx}$
- Polynomial functions $y = C_1 + C_2.x + C_3.x^2 + ...... + C_{n+1}.x^n$
- Trigonometric functions, and
- Probability distributions, etc.



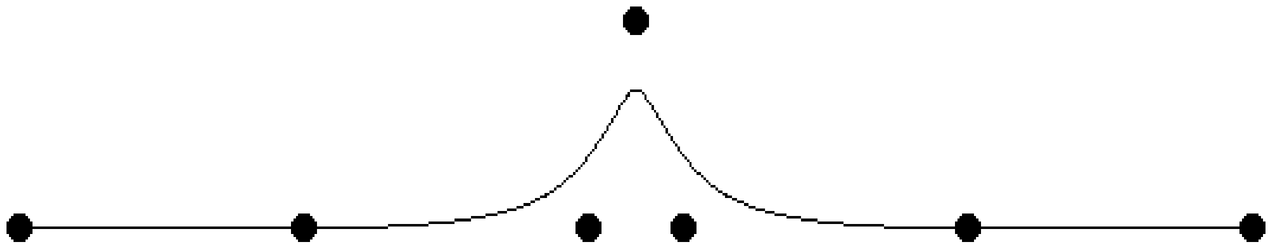

**Interpolation and Approximation of data points (a) curve fitting (b) curve fairing**

# INTERPOLATION AND APPROXIMATION…

- Interpolation Curve – over constrained → lots of (undesirable?) oscillations



- Approximation Curve – more reasonable?



# CLASSICAL REPRESENTATION OF CURVES

Mathematically, non-parametric and parametric equations are used for the representation of planar curves or space curves

## Non-parametric Curves

- A non-parametric curve representation may be *explicit* or *implicit*. In explicit form, coordinate(s) of a point $y$ and/or $z$ are explicitly represented as function of $x$.

  Explicit, non-parametric planar curve is represented as $y = f(x)$

  For example, equation of a straight-line $y = mx + b$. For each $x$ value, there is only one $y$ value

- Explicit, non-parametric space curve is represented as $x = x, \quad y = f(x) \; and \; z = f(x)$

- The above equation has one-to-one relationship. Therefore, this is not suitable for the representation of closed or multivalued curves.

- Closed or multiple valued planar curves, e.g., a circle, parabola, ellipse, etc. gives two values of $y$ for each value of $x$.

# CLASSICAL REPRESENTATION OF CURVES…

## Non-parametric Curves…

- This form of curve representation is known as *implicit* non-parametric form of the curve.
- Thus, a general implicit non-parametric planar curve can be represented as $f(x, y) = 0$
- For example, a general second-degree implicit non-parametric equation is written as

$$ax^2 + 2hxy + by^2 + 2gx + 2fy + c = 0$$

- Above equation gives a variety of two-dimensional (planar) curves called *conic sections*. The three forms of conic sections are parabola, hyperbola and ellipse.
- Circle is a special case of an ellipse whereas a straight line is obtained if coefficients

$$a = h = b = 0$$

- Depending upon the values of coefficients in the equation, a planar curve may be described by specifying the following conditions:

# CLASSICAL REPRESENTATION OF CURVES…

## Non-parametric Curves…

- ***Positions and slopes at the two endpoints of the curve segment***
- ***Positions of the two endpoints and slopes either at the beginning or at the end of curve segment***

➢ The implicit non-parametric representation for space curves may be obtained by the intersection of two surfaces $f(x, y, z) = 0$ and $g(x, y, z) = 0$

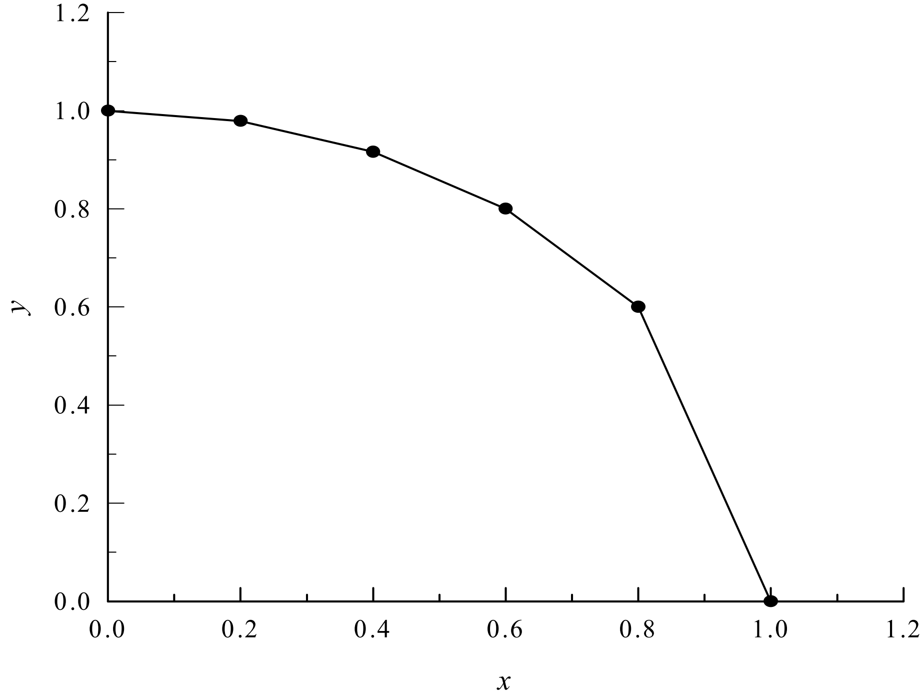
**Properties**

I. Explicit and implicit non-parametric curve representations are *axis dependent*.

II. A non-parametric representation of curve results into unequal distribution of points on the curve, which in turn, affects the ***quality*** and ***accuracy*** of the curve.

III. If a curve is to be displayed as a series of points or straight-line segments, the computations involved could be expensive.

# CLASSICAL REPRESENTATION OF CURVES…




**Explicit non-parametric representation of a unit radius origin centered circle**

# PARAMETRIC CURVES

- *Parametric* representations of closed or multivalued curves overcome the difficulties associated with the non-parametric representations.
- Parametric representations for commonly used curves such as conic sections employ *polynomials* in place of equations involving the *square root calculations*.
- Thus, parametric representations for the curves are more general and suitable for the CAD applications due to the *ease in computations*.
- **In parametric form, each point on the curve is expressed as a function of single parameter**.
- Thus, position vector of a point on the curve is fixed by a single parameter.

For two-dimensional (planar) curve with $t$ as a parameter, the Cartesian coordinates of a point on the curve is expressed as

$$x = x(t) \text{ and } y = y(t) \quad \text{or} \quad P(t) = \begin{Bmatrix} x(t) \\ y(t) \end{Bmatrix}$$

- A single non-parametric curve equation (in terms of $x$ and $y$) may be obtained from two parameter equations by eliminating the parameter $t$.
- The *tangent vector* (or derivative) for a parametric curve is defined as

$$P'(t) = \begin{Bmatrix} x'(t) \\ y'(t) \end{Bmatrix}$$

- Therefore, the slope of the parametric curve is given by

$$\frac{dy}{dx} = \frac{dy/dt}{dx/dt} = \frac{y'(t)}{x'(t)}$$

- With parametric representations, the infinite slope condition ( $dy/dx = \infty$ ) can easily be obtained by substituting $x'(t) = 0$
- Since a point on the curve is specified by a single parameter $t$; therefore, the parametric curves are *axis independent*. Mostly $0 \le t \le 1$
- The position and slope at the endpoints of the curve is specified by the parameter $t$, which is fixed within the parameter range.

# PARAMETRIC RERESENTATION OF ANALYTIC CURVES

## Lines

The parametric representation of a straight line connecting the two position vectors $P_1$ and $P_2$ is given by

$$P(t) = P_1 + (P_2 - P_1).t \quad , \qquad 0 \le t \le 1$$

The position vector $P\,(t)$ has a parametric representation $x\,(t)$ and $y\,(t)$ ; therefore

$$x(t) = x_1 + (x_2 - x_1).t \qquad 0 \le t \le 1$$

$$y(t) = y_1 + (y_2 - y_1).t \qquad 0 \le t \le 1$$

Moreover, tangent vector of the line is given as

$$P'(t) = P_2 - P_1$$

where

$$x'(t) = x_2 - x_1$$

$$y'(t) = y_2 - y_1$$

Thus, the *tangent vector of the line is independent of the parameter t*. The infinite slope (vertical line) condition and zero slope (horizontal line) condition can be obtained from above eqn.

# PARAMETRIC RERESENTATION OF ANALYTIC CURVES

## Circles

The parametric equation of origin centered circle in standard trigonometric form is given as

$$\left.\begin{aligned} x &= a.\cos\theta \\ y &= a.\sin\theta \\ z &= 0 \end{aligned}\right\} \quad 0 \le \theta \le 2\pi$$

where parameter θ is the angle measured in ccw direction from the positive *x*–axis. The parametric equation of non-origin centered circle is given as

$$\left.\begin{aligned} x &= x_c + a.\cos\theta \\ y &= y_c + a.\sin\theta \\ z &= z_c \end{aligned}\right\} \quad 0 \le \theta \le 2\pi$$

where coordinates $(x_c, y_c, z_c)$ is the centre of the circle.

- The parametric representation for a curve is not unique.
- A computationally less expensive parametric representation technique uses the polynomials for representing the curves.

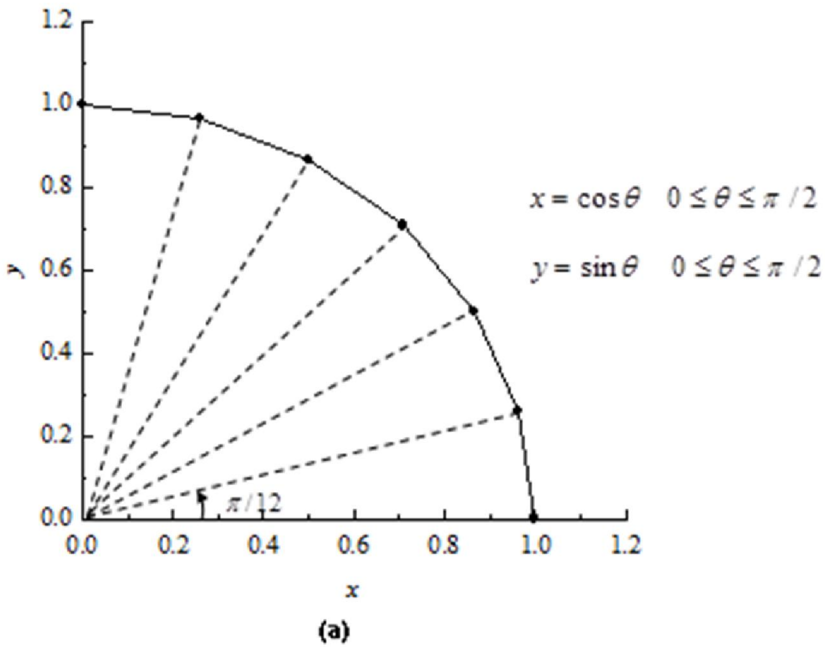
## Circles…

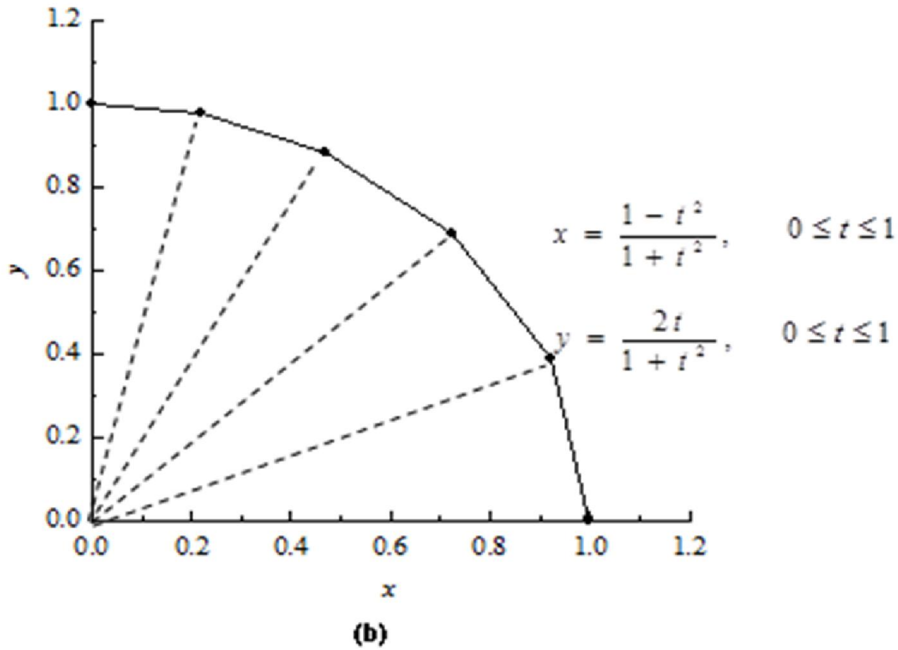
The polynomial form of parametric representation of a circle is expressed as

$$x = \frac{1-t^2}{1+t^2} \quad \text{and} \quad y = \frac{2t}{1+t^2} \qquad 0 \le t \le 1$$

- Although, this results into unequal perimeter lengths on the circumference of a circle for equal increments in parameter $t$ but the quality of circle is much better than the explicit nonparametric representation.
- However, the quality of circle obtained with polynomial representation is not as good as the standard trigonometric representation.
- Polynomial form of parametric representation is computationally less expensive and it may be considered as a compromise between the quality and computations.

## Circles…




**Parametric Representations of circles in first quadrant**

**(a) Trigonometric (b) Polynomial**

# PARAMETRIC RERESENTATION OF ANALYTIC CURVES

## Ellipse

The parametric representation of non-origin centered ellipse is given as

$$\left.\begin{aligned} x &= x_c + a.\cos\theta \\ y &= y_c + b.\sin\theta \\ z &= z_c \end{aligned}\right\} \quad 0 \le \theta \le 2\pi$$

## Parabola

In rectangular coordinates, the non-parametric representation of origin centered parabola opening to the right is

$$y^2 = 4ax$$

A parametric representation, in trigonometric form, is given as

$$y = \pm 2\sqrt{a.\tan\theta} \qquad 0 \le \theta \le \pi/2$$

An alternative parametric representation, in polynomial form, is given as

$$\begin{aligned} x &= x_v + at^2 \\ y &= y_v + 2at \qquad 0 \le t \le \infty \\ z &= z_v \end{aligned}$$

## Hyperbolas

The parametric representation of hyperbola is given as

$$\left.\begin{aligned} x &= x_v \pm a\sec\theta \\ y &= y_v \pm b\tan\theta \\ z &= z_v \end{aligned}\right\} \quad 0 \le \theta \le \pi/2$$

# COMPUTER AIDED DESIGN
## (BME-42)

## Unit-III: Space Curves

**(7 Lectures)**

- **Properties for curve design, Parametric continuity,**
- **Parametric representation of synthetic curves, Spline curves and specifications**, **Parametric representation of synthetic curves**
- Hermite curves-Blending functions formulation, shape control, properties,
- Bezier curves-Blending functions formulation, properties, Composite Bezier curves,
- Non-rational B-spline curves- Blending functions formulation, knot vector, B-spline blending functions, properties

# Lecture 20

## Topics Covered

**Space Curve**
**Properties for Curve Design**
**Parametric Representation of Synthetic Curves**
**Spline Curves**
**Specifications of Spline Curves**
**Classification of Spline Curves**
Hermite Curves, Bezier Curves, B-Spline Curves

*Prepared By*

Prof. S. K. SRIVASTAVA

MED, MMMUT, Gorakhpur (UP)
sksme@mmmut.ac.in

# SPACE CURVES

- Space (three-dimensional) curves and surfaces are mostly used in the design of *automobile bodies, aerospace wings, ship hulls, propeller blades, shoes, bottles*, etc.
- These applications require *curves* and *surfaces* as basic entities.
- Curve is the collection of points and they form basic entities of the surfaces. Surfaces can be obtained by digitizing the physical model or a drawing, followed by curve fitting through the set of data points.
- Mathematically, curve ***fitting*** (interpolation) and curve ***fairing*** (approximation) techniques are used for generating the curves in computer graphics.
- The analytical form of planar curves is not suitable for designing the complex three-dimensional curves and surfaces used for designing the complex shaped objects.
- The designer prefers the synthetic curve, which passes through the set of data points, because designer has full control on its shape as per the new design requirements.

# PROPERTIES FOR CURVE DESIGN

- In computer graphics, a curve is represented in such a manner that it must be mathematically tractable and computationally convenient.
- The experiences of designer suggest that the curve must possess the following important properties for the design and representation in computer graphics:

## Control points

- The control points govern the shape of the curve in a *predictable* manner.
- It is possible to control the shape of the curve *interactively* through proper location of the control points.
- A curve must interpolate (pass) the control points.

## Axis Independence

- The shape of curve must not change if control points are measured in different coordinate system.

# PROPERTIES FOR CURVE DESIGN

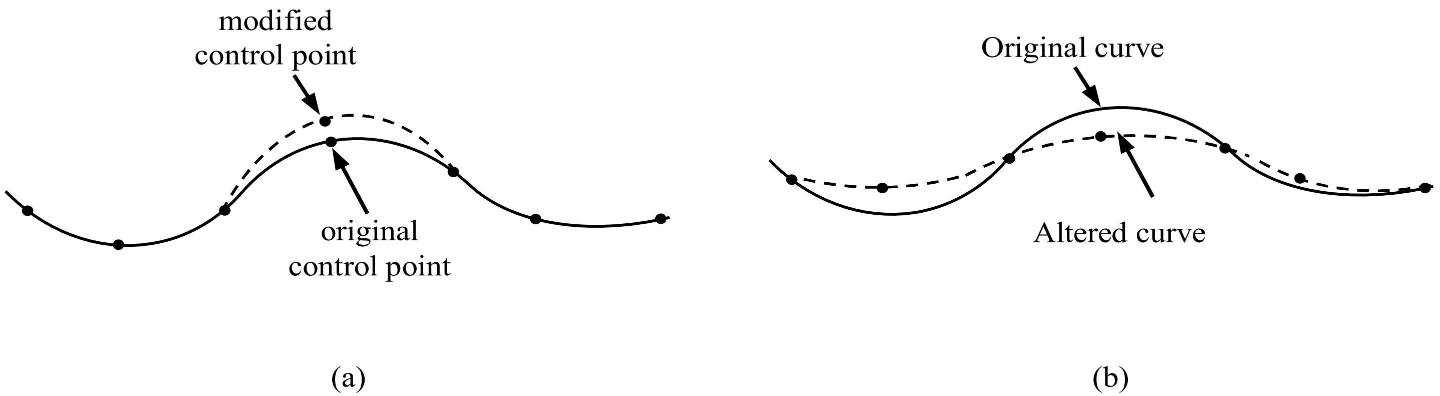
**Axis Independence…**

- For example, if control points rotate by $30^o$, the entire curve must rotate by $30^o$, keeping the *shape unchanged*.
- Due to its axis independent nature, it is possible to transform a parametric curve into a curve of the same shape but with different orientations.

**Local Control and Global Control**

- In computer graphics, it is frequently required to modify the portion of the curve.
- The curve may change its shape only in the portion near to the control point or the shape of entire curve may change.
- The first modification in the curve shape is termed *local control* whereas second modification as *global control*.
- A designer is always interested in *local control* because altering the position of control point *does not propagate the change in the remaining portion of the curve*.

## Local Control and Global Control…




**Shape control of curve (a) Local (b) Global**

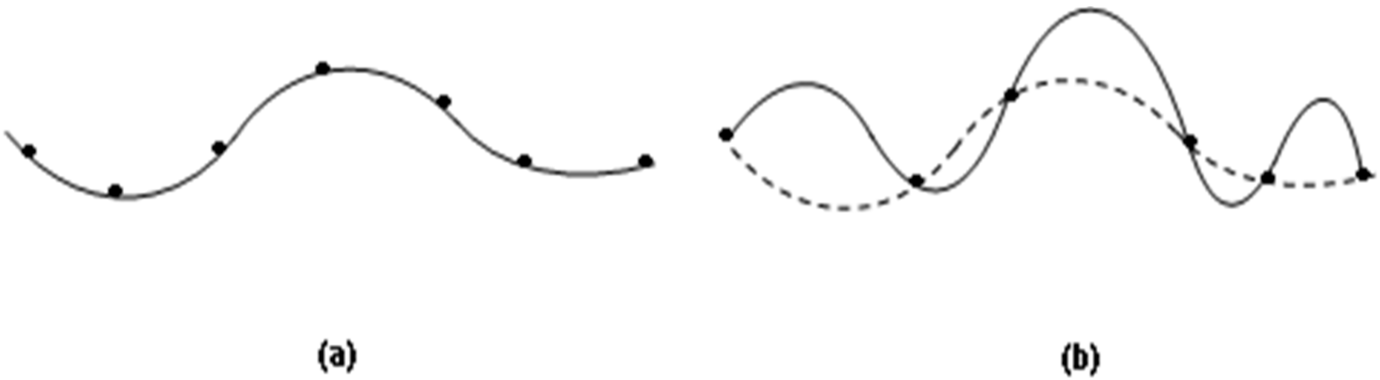
## Variation Diminishing Property

- A curve is said to be smooth if it has a tendency to pass through the control points *smoothly*.
- If curve *oscillates* about the control points it is usually *not desirable*.

# PROPERTIES FOR CURVE DESIGN

## Variation Diminishing Property…

- Thus, the curve which passes through the control points smoothly and does not show the tendency to amplify the irregularities in the form of oscillations, is said to possess the variation diminishing property.




**(a) Variation diminishing property**

**(b) Curve with undesirable oscillations about the control points**

## Versatility

- The mathematical model for curve representation should allow the designer to change its shape by either adding or removing the control points..

### Versatility…

- This implies the versatility of the curve, i.e., addition of the control points defining the curve gives *additional shapes* to the curve depending upon the position of additional control points.

### Order of Continuity

- It is difficult to achieve the complex shape of object with a single curve.
- Usually, several curves are joined together end to end to accomplish the complex shape.
- The order of continuity decides the exact shape of the joint.
- The parametric continuity results by matching the *parametric positions* and the *parametric derivatives* of adjoining curves at their common boundary.

There are three types of order of continuity:

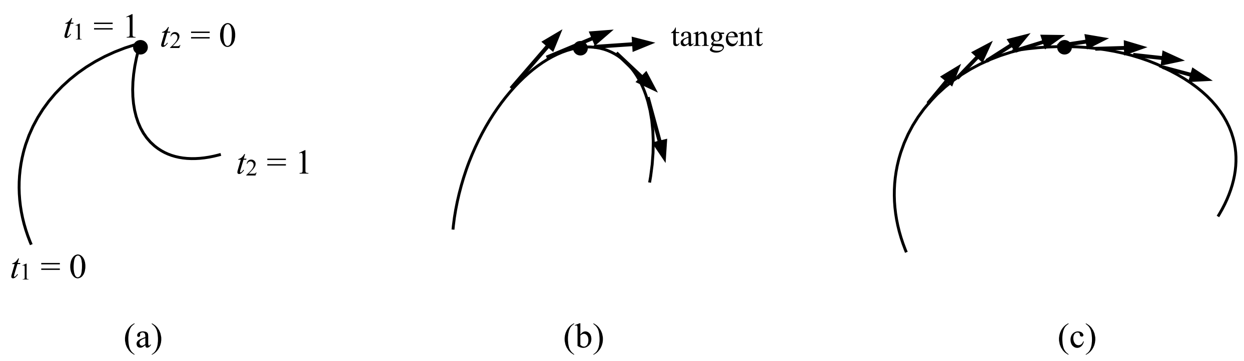
#### *Zero Order Continuity or $C^0$ Continuity*

Zero order continuity exists when adjoining curves simply meet to form a joint, i.e., parameter $t$ at the joint, for the two adjoining curves are same. Figure (a) shows the zero order (position) continuity.

## Order of Continuity…

### *First Order Continuity or $C^1$ Continuity*

- First order continuity exists when first order derivatives (i.e., tangent) for two adjoining curves, at their joining point, are same.
- The rate of change of the tangent vectors (second derivatives) can be quite different so that general shapes of the two adjacent sections can change abruptly.
- A joint with continuity also possesses $C^0$ and $C^1$ continuity. Figure (b) shows the first order (tangent) continuity.




**Parametric continuities at the junction point of two curves**

**(a) Zero Order Continuity (b) First Order continuity (c) Second Order Continuity**

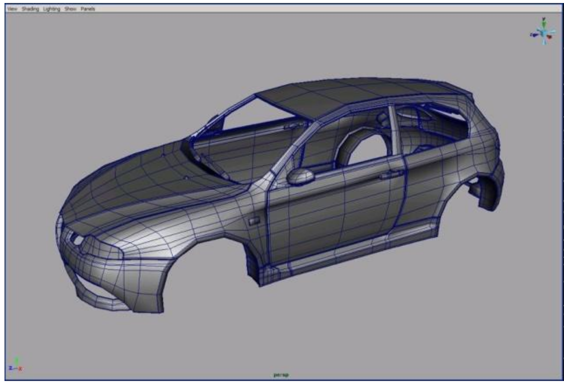
## Order of Continuity…

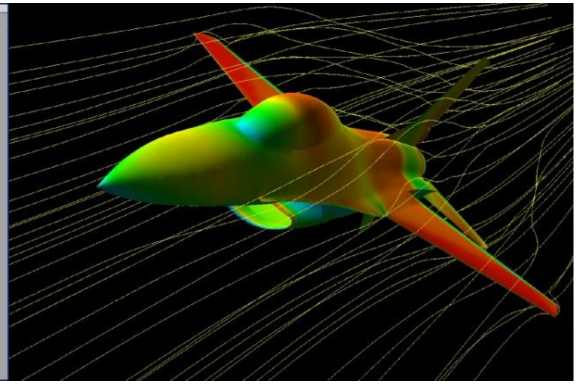
### *Second Order Continuity or $C^2$ Continuity*

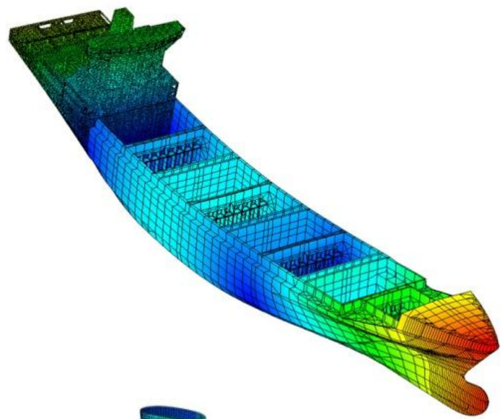
- First order continuity is generally sufficient for *digitizing the drawings* whereas second order continuity is required for setting up *animation paths for the camera motion*.
- A camera moving in $C^1$ continuity or tangent continuity with equal steps in parameter $t$ experiences acceleration at the joint of two adjoining curve sections, leading to the discontinuity in motion in the form of jerks.
- Therefore, $C^2$ continuity or *curvature continuity* is desirable for camera motion during the animation.

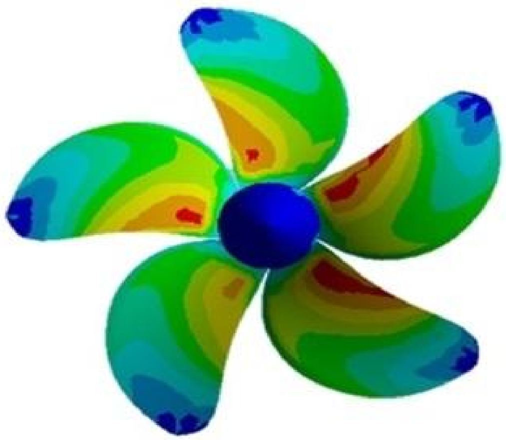
# SYNTHETIC CURVES

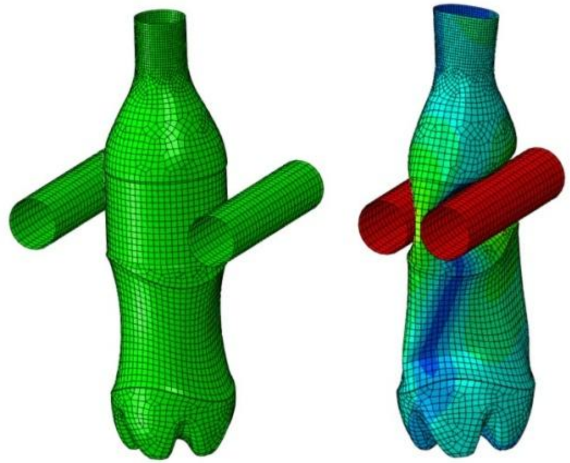
Space (three-dimensional) curves are mostly used in the design of **automobile bodies, aerospace wings, ship hulls, propeller blades, shoes, bottles**, etc.











# PARAMETRIC REPRESENTATION OF SYNTHETIC CURVES

Synthetic curves are highly suitable for defining the complex curves and surfaces. The designer, mainly in the following situations, prefers them:

- When the space curve is represented with the help of data points (control points) and its shape changes by shifting the data points (defining the curve) to meet the new design requirement.
- When a designer needs the space curve representation in such a way that shifting of one or more data points, changes its shape by the processes of *twisting* (non-planar), *bending* and *stretching*.

➢ Mathematically, synthetic curve generation is a curve fitting/fairing problem wherein smooth curves generate through a set of known measured data points.

➢ The order of continuity is very important for generating a complex shape smooth curve because it is modeled by joining several curve segments end to end.

# PARAMETRIC REPRESENTATION OF SYNTHETIC CURVES…

- Typically, synthetic curves are represented in *polynomial* forms because it is easy to apply various orders of continuity conditions such as position $(C^0)$, slope $(C^1)$ and curvature $(C^2)$ at the boundaries of small curve segments.
- Polynomials are easy to *differentiate/integrate*; therefore, speed up the computation process.

In polynomial form, the $x$ variation of parameter for the first, second and third order polynomials, respectively, may be expressed as

| | | |
|---|---|---|
| First order (line segment) | : | $x(t) = at + b$ |
| Second order (quadratic polynomial) | : | $x(t) = at^2 + bt + c$ |
| Third order (cubic polynomial) | : | $x(t) = at^3 + bt^2 + ct + d$ |

# PARAMETRIC REPRESENTATION OF SYNTHETIC CURVES…

- The lowest degree polynomials such as first order polynomial requires the determination of two coefficients *a* and *b* which can be calculated from the endpoint coordinates (two boundary conditions) of the line segment.

- For the line segment, the line and slope (derivative) at the endpoints are same; therefore, the endpoints coordinates, and tangents (slopes) at the endpoints, cannot control the shape of curve (line), independently.

- With quadratic polynomials, three coefficients *a*, *b* and *c* can be calculated, using two endpoint conditions and one boundary condition such as slope (tangent) at one endpoint or one additional point outside the curve, which controls the tangent at the endpoint.

- If three points describe the quadratic polynomial then polynomial lies in a plane defined by the three points; therefore, interpolation becomes difficult.

# PARAMETRIC REPRESENTATION OF SYNTHETIC CURVES…

- A *cubic* polynomial is the lowest degree polynomial, which generates the curve with $C^0$, $C^1$ and $C^2$ continuities.
- The cubic polynomials represent a *non-planar* (twisted) space curve.
- The curves and surfaces with higher degree polynomials experience oscillations about the control points.
- The higher degree polynomials are computationally expensive and require large amount of storage.
- However, they are preferred in the design of car bodies and aerospace and hydrospace structures because aerodynamically efficient shape requires the control of *higher degree derivatives* at the boundaries of the curve/surface segments.

# SPLINE CURVES

- Physically, spline is a long flexible strip of metal/plastic/wood used to produce the curve through the known set of data points.

- The curved shape of the strip is obtained by pulling it into the transverse direction using the lead weights or pegs. The lead weights or pegs hold the strip into the curved position.

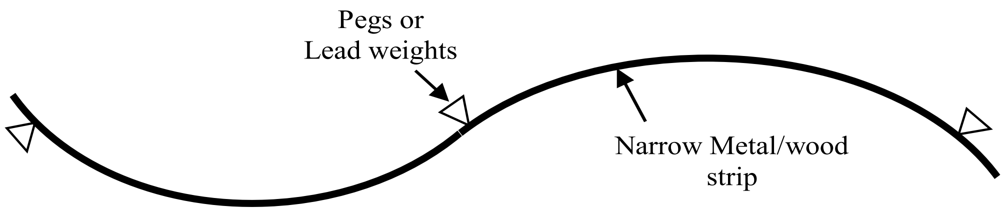
- The spline shape of the strip can be obtained by varying the number of lead weights and its positions on the board by the drafters.

- The resulting curve appears smooth and fits the pegs (data points).

- The term *spline curve* was originally referred to a curve drawn in this manner by a drafter. This spline curve is a natural cubic spline possessing $C^2$ continuity.

# SPLINE CURVES…

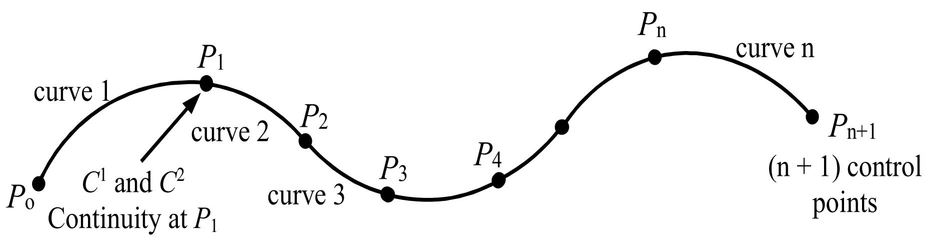
**Physical Spline Shape**




# SPLINE CURVES…

- The curve fitting for the $(n+1)$ control points require $n$ curve segments.
- A parametric cubic spline $x(t) = at^3 + bt^2 + ct + d$ has four coefficients.
- Thus, a total of $4n$ coefficients are required for the $n$ curve segments.
- The main problem with the natural or drafting spline is the local control.
- If any one of the control point shifts, the entire curve is affected; therefore, does not allow local control for the natural spline curves.
- The designer does not prefer the natural cubic spline because it is not possible to restructure part of the curve without affecting the entire curve.

**A piecewise $C^2$ continuous cubic spline interpolation of ($n$+1) control points**



In modern computer graphics, splines are preferred for the following applications:

- Design of various types of curves
- Design of surface shapes
- Digitization of drawings for the computer storage
- Specification of animation paths for the camera or eyes
- Design of aerodynamic efficient automobile bodies
- Design of aerospace structures such as surface of aeroplanes, rockets, etc.
- Design of hydrospace structures such as surface of ship hulls, submarines, etc.
- Design of curved shape products such as shoes, bottles, etc.

## Specifications of spline curves

There are three methods for characterizing the spline curves:

- **On the basis of set of imposed boundary conditions**
- **On the basis of a matrix that characterizes the specific spline curve**
- **On the basis of blending (or basis) functions that characterizes the spline curve**

These three spline specifications may be illustrated by assuming the cubic polynomial parametric representation, for *x*, *y* and *z* coordinates along the spline path in parameter *t* as

$$\left.\begin{aligned} x(t) &= a_x.t^3 + b_x.t^2 + c_x.t + d_x \\ y(t) &= a_y.t^3 + b_y.t^2 + c_y.t + d_y \\ z(t) &= a_z.t^3 + b_z.t^2 + c_z.t + d_z \end{aligned}\right\} \quad 0 \le t \le 1$$

In matrix form, we have

$$\{x(t) \quad y(t) \quad z(t)\} = \begin{bmatrix} t^3 & t^2 & t & 1 \end{bmatrix} \begin{Bmatrix} a_x & a_y & a_z \\ b_x & b_y & b_z \\ c_x & c_y & c_z \\ d_x & d_y & d_z \end{Bmatrix}$$

## Specifications of spline curves…

or $P(t) = T.C$

where $C$ is the *polynomial coefficient matrix*. For each polynomial $x\ (t)$, $y\ (t)$ and $z\ (t)$, it is required to calculate four coefficients $a$, $b$, $c$, and $d$.

- Therefore, four boundary conditions are required for the four unknown coefficients.
- These constants are evaluated by imposing the sufficient boundary conditions at the junction of two curve segments

The boundary conditions may be:

- **Constraints (positions) at the endpoints**
- **Tangents (slopes) at the endpoints**
- **Continuity at the junction between the curve segments**

# CLASSIFICATIONS OF SPLINE CURVES

Based upon the techniques for the evaluation of four coefficients *a*, *b*, *c* and *d*, there are three major classifications of spline curves:

## *Hermite Curves*

- Hermite curve is also known as *cubic curve* characterized by the *two endpoints* and *tangent vectors* at the endpoints.
- Hermite curve passes (interpolate) through the endpoints of the curve segment and possesses first order (slope) continuity.

## *Bézier Curves*

- *Two endpoints* and *two additional points* outside the curve characterize the Bézier curves.
- The additional points outside the curve control the endpoints tangent vectors.

# CLASSIFICATIONS OF SPLINE CURVES

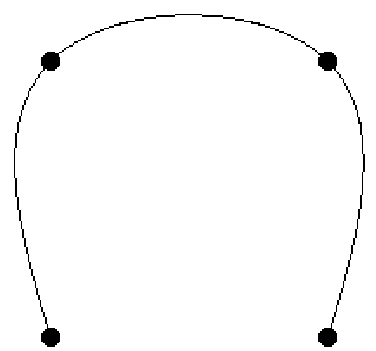
### *Bézier Curves…*

- Thus, Bézier curve interpolates the endpoints and approximates the additional points outside the curve, i.e., they do not pass through the outside points.
- Bezier curves also possess first order (slope) continuity.

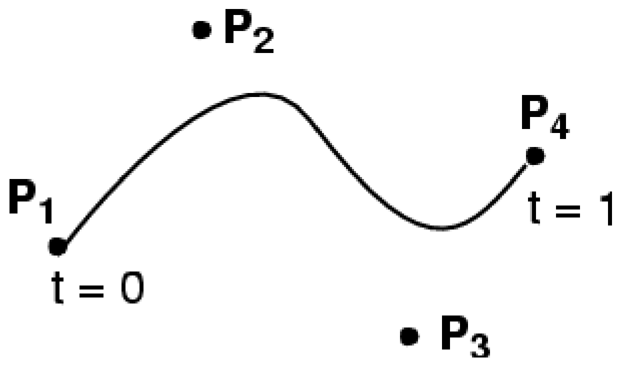
### *B-spline Curves*

- B-spline curves are characterized by approximating the endpoints, allowing first and second order derivatives ($C^1$ and $C^2$ continuity) to be continuous at the endpoints of the curve.
- Under certain conditions, the curve may interpolate the endpoints.

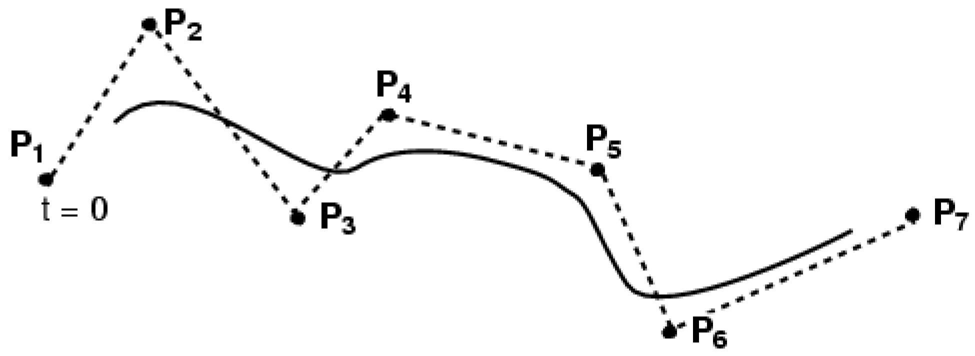
# CLASSIFICATIONS OF SPLINE CURVES



Interpolation (Hermite curve)




Approximation (Bézier curve)




Approximation (B-Spline curve)

# COMPUTER AIDED DESIGN
## (BME-42)

### Unit-III: Space Curves

**(7 Lectures)**

- Properties for curve design, Parametric continuity,
- Parametric representation of synthetic curves, Spline curves and specifications, Parametric representation of synthetic curves
- **Hermite curves-Blending functions formulation, shape control, properties**,
- Bezier curves-Blending functions formulation, properties, Composite Bezier curves,
- Non-rational B-spline curves- Blending functions formulation, knot vector, B-spline blending functions, properties

# Lecture 21

## Topics Covered

**Hermite Curves**

**Blending Function Formulation**

**Shape Control**

**Effect of Continuities on the Shape**

**Limitations**

Example

*Prepared By*

**Prof. S. K. SRIVASTAVA**

MED, MMMUT, Gorakhpur (UP)
sksme@mmmut.ac.in

# HERMITE CURVES

A Hermite (named after the mathematician *Charles Hermite*) or Ferguson's cubic curve is an interpolating piecewise cubic polynomial having specified tangents at each end control points.

Thus,

- Hermite curve is also known as *cubic* curve
- Characterized by the *two endpoints* and *tangent vectors* at the endpoints.
- Passes (*interpolates*) through the endpoints of the curve segment and possesses *first order (slope)* continuity at the endpoints.

➢ Unlike the natural cubic splines, Hermite curve segments adjusted locally because shape of each curve segment is dependent only upon its endpoint constraints.

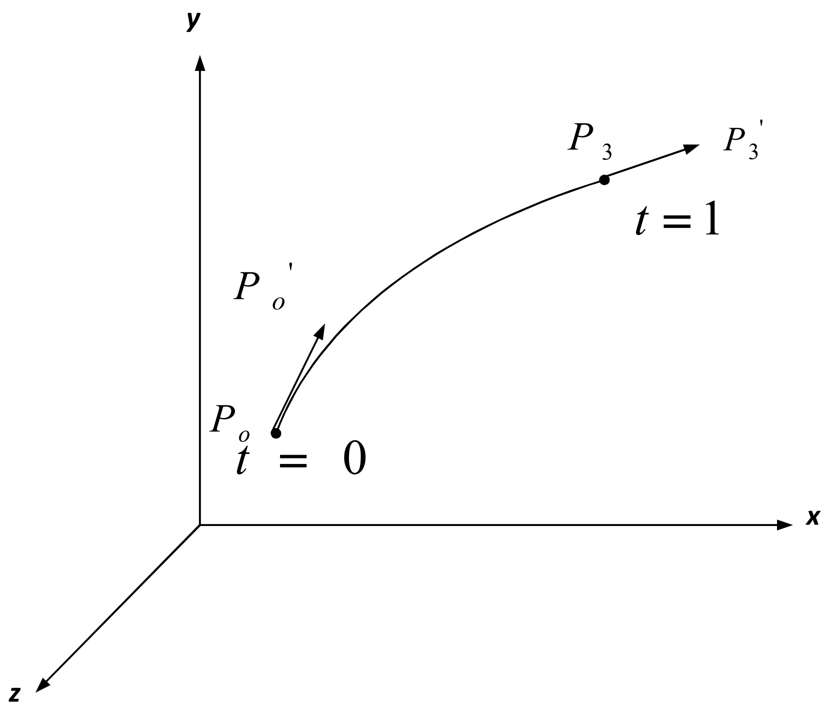
## Blending Function Formulation

- Let $P_o$ and $P_3$ are the point *position vectors*, and $P_o'$ are $P_3'$ the *tangent vectors* at the two endpoints of the cubic curve segment.
- It is required to find out a relationship among the following matrices

- Hermite basis matrix, $M_H$
- Hermite geometry matrix, $G_H$
- Polynomial coefficient matrix, $C_x$

Where $x$ component of Hermite geometry matrix ($G_H$) and polynomial coefficients matrix ($C_x$) may be expressed as

$$G_{Hx} = \begin{Bmatrix} P_o \\ P_3 \\ P_o' \\ P_3' \end{Bmatrix}_x \quad \text{and} \quad C_x = \begin{Bmatrix} a \\ b \\ c \\ d \end{Bmatrix}_x$$



## Blending Function Formulation…

The cubic polynomial equation for $x$ ($t$) may be expressed as

$$x(t) = a_x.t^3 + b_x.t^2 + c_x.t + d_x$$

$$= \begin{bmatrix} t^3 & t^2 & t & 1 \end{bmatrix} \begin{Bmatrix} a \\ b \\ c \\ d \end{Bmatrix}_x$$

$$x(t) = T.C_x$$

For Hermite curve, the *boundary conditions* are

I. Position of endpoints

at $t = 0$, $x(0) = P_{ox} = \begin{bmatrix} 0 & 0 & 0 & 1 \end{bmatrix}.C_x$

at $t = 1$, $x(1) = P_{3x} = \begin{bmatrix} 1 & 1 & 1 & 1 \end{bmatrix}.C_x$

**Blending Function Formulation…**

II. Tangent vector (first derivative) at the endpoints is given as

$$x'(t) = \begin{bmatrix} 3t^2 & 2t & 1 & 0 \end{bmatrix} C_x$$

Therefore,

at $t = 0$, $\quad x'(0) = P'_{ox} = \begin{bmatrix} 0 & 0 & 1 & 0 \end{bmatrix} C_x$

at $t = 1$, $\quad x'(1) = P'_{3x} = \begin{bmatrix} 3 & 2 & 1 & 0 \end{bmatrix} C_x$

In matrix form, above four eqns. may be combined as

$$\begin{Bmatrix} P_o \\ P_3 \\ P'_o \\ P'_3 \end{Bmatrix}_x = \begin{bmatrix} 0 & 0 & 0 & 1 \\ 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 0 \\ 3 & 2 & 1 & 0 \end{bmatrix} .C_x \quad \text{or} \quad C_x = \begin{bmatrix} 2 & -2 & 1 & 1 \\ -3 & 3 & -2 & -1 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix} . \begin{Bmatrix} P_o \\ P_3 \\ P'_o \\ P'_3 \end{Bmatrix}_x$$

or $\quad C_x = M_H . G_{Hx}$

## Blending Function Formulation…

Thus, the cubic polynomial equation for $x(t)$ may be expressed as

$$x(t) = T.C_x = T.M_H.G_{Hx}$$

Similarly, $$y(t) = T.M_H.G_{Hy}$$

$$z(t) = T.M_H.G_{Hz}$$

Thus, a point on the Hermite curve is defined as

$$P(t) = \{x(t) \quad y(t) \quad z(t)\} = T.M_H.G_H$$

Now, the expansion of $x(t)$ gives

$$x(t) = T.M_H.G_{Hx} = [t^3 \quad t^2 \quad t \quad 1]\begin{bmatrix} 2 & -2 & 1 & 1 \\ -3 & 3 & -2 & -1 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix} . \begin{Bmatrix} P_o \\ P_3 \\ P_o^{'} \\ P_3^{'} \end{Bmatrix}_x$$

$$x(t) = P_{ox}(2t^3 - 3t^2 + 1) + P_{3x}(-2t^3 + 3t^2) + P_{ox}^{'}(t^3 - 2t^2 + t) + P_{3x}^{'}(t^3 - t^2)$$

## Blending Function Formulation…

or $$x(t) = P_{ox}.H_o(t) + P_{3x}.H_1(t) + P'_{ox}.H_2(t) + P'_{3x}.H_3(t)$$

Similarly $$y(t) = P_{oy}.H_o(t) + P_{3y}.H_1(t) + P'_{oy}.H_2(t) + P'_{3y}.H_3(t)$$

$$z(t) = P_{oz}.H_o(t) + P_{3z}.H_1(t) + P'_{oz}.H_2(t) + P'_{3z}.H_3(t)$$

Thus, $$P(t) = P_o.H_o(t) + P_3.H_1(t) + P'_o.H_2(t) + P'_3.H_3(t)$$

The above equation represents ***blending function formulation*** of Hermite curve.

The polynomial $H_k(t)$ where $k$ = 0, 1, 2, 3 are referred to as ***Hermite blending functions*** because they blend (control) the boundary constraints (endpoints $P_o$ & $P_3$ and tangent vectors at the endpoints $P'_o$ & $P'_3$ ), used to calculate the coordinate positions on the Hermite curve.

- Figure shows the effect of four Hermite blending functions $H_o(t)$ $H_1(t)$ $H_2(t)$ and $H_3(t)$ on the coordinate positions along the curve.

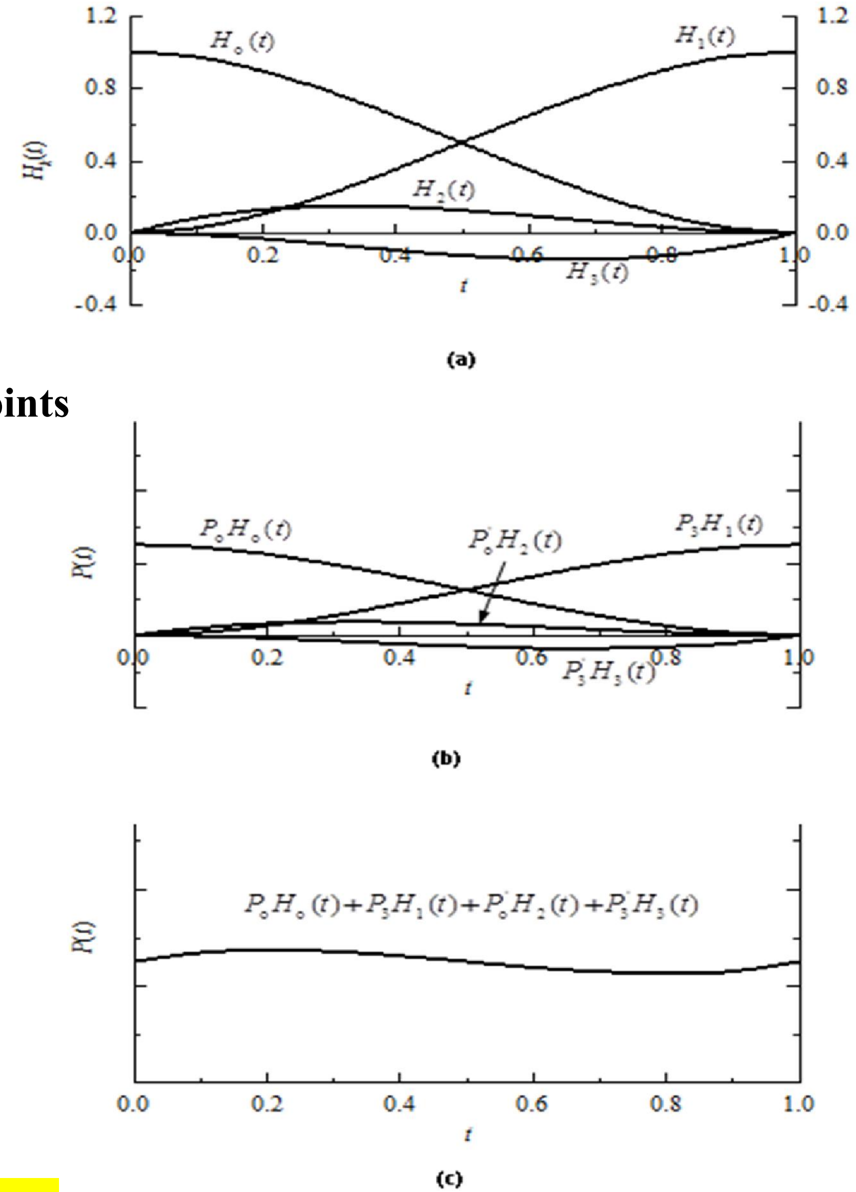
## Blending Function Formulation…

**Generation of Hermite cubic curve**

**(a) Effect of blending functions**

**(b) Effect of positions and slopes at the endpoints**

**(c) Hermite cubic curve**



## Blending Function Formulation…

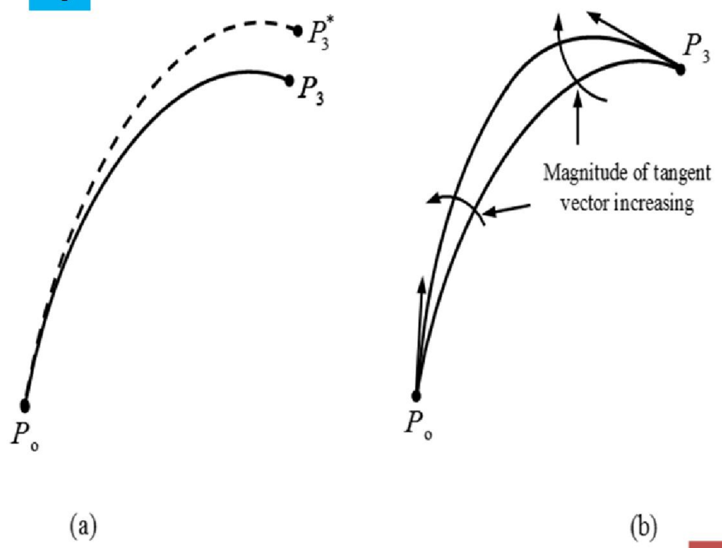
- At $t = 0$, the blending function that controls the endpoint, i.e., $H_o(t) \neq 0$; hence, it affects the shape of the curve.
- As parameter $t$ increases, the other blending functions $H_1(t)$, $H_2(t)$ and $H_3(t)$ begins to *influence the curve*.
- It should be noted that the effect of $H_3(t)$ is negative.
- Figure shows the four functions $H_o(t)$, $H_1(t)$, $H_2(t)$ and $H_3(t)$ *weighted* by the $x$ components of geometry vector $P_{ox}$ and $P_{3x}$, and tangent vectors at the endpoints $P'_{ox}$ and $P'_{3x}$.
- It is obvious that the effect of endpoint tangent vectors have *less influence* than the end position vectors on the shape of Hermite curve segment.
- Figure shows the sum of these blending functions resulting into Hermite curve.

## Shape Control

There are three ways to modify/control the shape of Hermite (cubic) curves:

1. Change in the location of the control points
2. Change in the magnitudes of the tangent vectors, keeping the same directions
3. Change in the directions of the tangent vectors, keeping the same magnitudes

1. Figure (a) shows the change in the shape of Hermite curve wherein the position of control point $P_3$ has been shifted to the new location $P_3^*$




**Shape control of Hermite curve**

**(a) change in location (position) of control point**

**(b) change in magnitude of tangent vector (slope)**

## Shape Control …

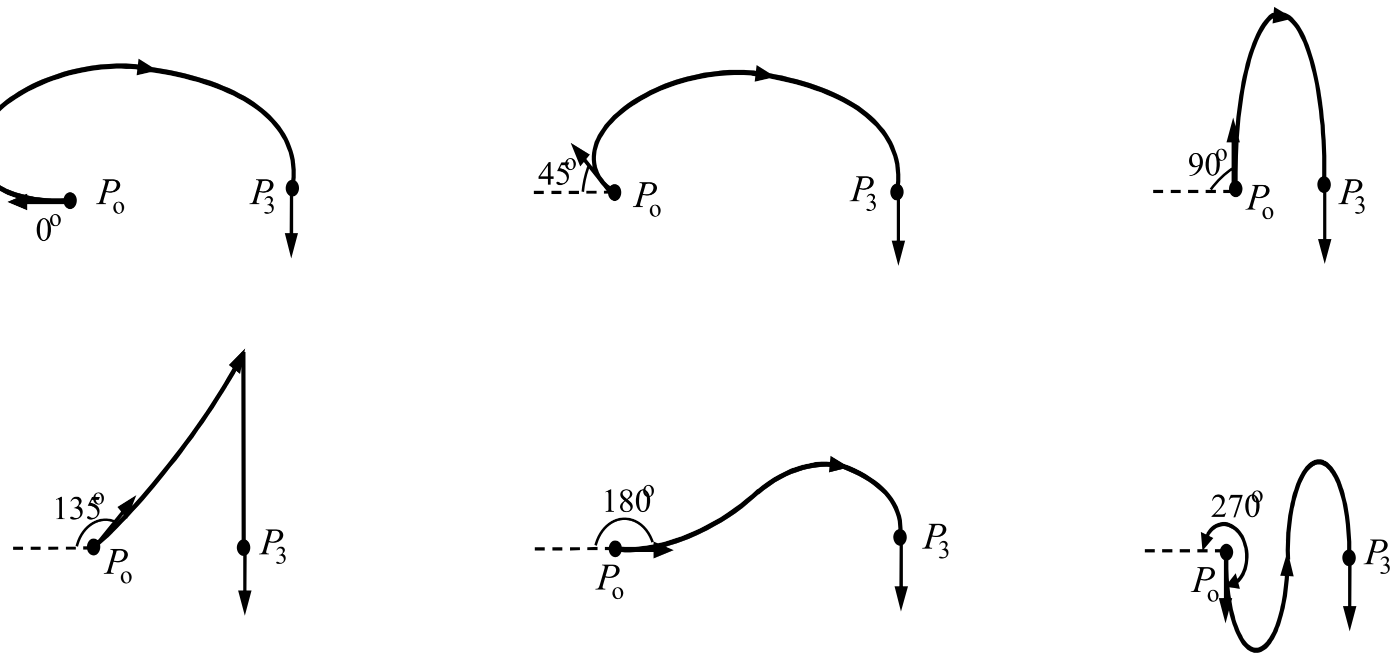
2. Figure (b) shows the shape control of Hermite curves when the magnitude of tangent vectors is changed at point $P_o$ keeping the same directions at the ends of the curve.

- It is observed that *longer the tangent vectors, the greater their effect on the curve*.
- From Figure (b), it has been concluded that the *effect of endpoint tangent vectors* have *less influence than the endpoints position vectors* on the shape of Hermite curves, but the effect can still be significant.

3. Figure (c) shows series of Hermite curves depicting only the effect of change in direction of the tangent vectors at the starting point $P_o$ on the shape of cubic curves.

All tangent vectors have the same magnitudes (lengths) either at $P_o$ or $P_3$

## Shape Control …

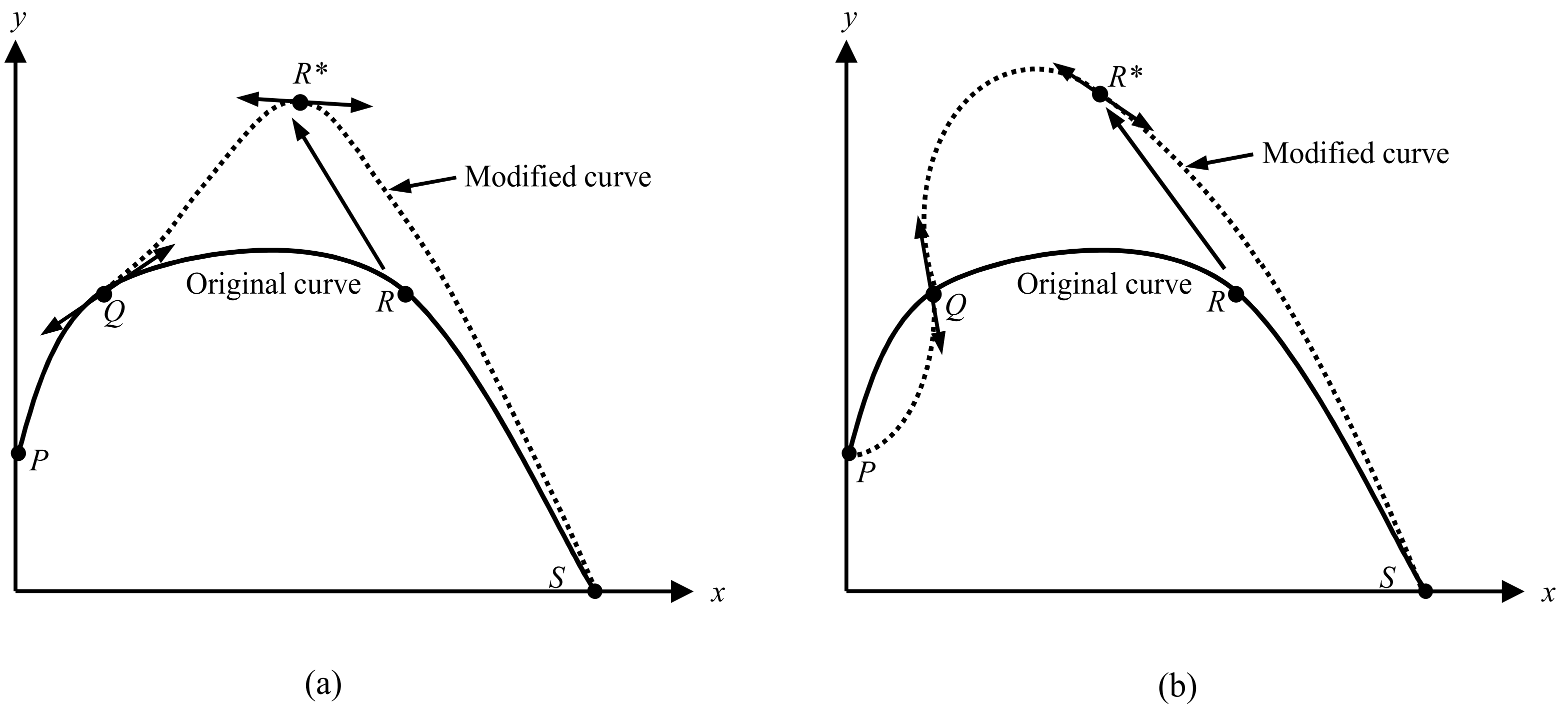



**(c) Effect of tangent vector directions keeping constant magnitude at point $P_3$ on the shape of family of Hermite curves**

## Effect of Continuities on the Shape …

- Figures show the effects of $C^1$ and $C^2$ continuities on the shape of Hermite curves.
- When a control (data) point ($R$) in a $C^1$ continuous composite Hermite curve is shifted to the new location ($R^*$) a change in the shape of maximum of two curve segments on either side of the shifted control point ($R^*$) occurs.
- This results into change in the slope of curve at the junction point; consequently, changes the shape of Hermite curve as shown in Figure (a).
- Alternatively, composite Hermite curves with $C^1$ continuity possess the *local shape control properties.*
- However, altering a control point in $C^2$ continuous composite curves possess a *global shape control properties* as shown in Figure (b).
- Thus, all curve segments are affected.

## Effect of Continuities on the Shape…




**Effect of continuities on the shape of Hermite curves**

**(a) original and modified $C^1$ continuous curve depicting local change in the curve shape**

**(b) original and modified $C^2$ continuous curve depicting global change in the curve shape**

## Limitations

- ❖ Hermite curves are suitable for some *digitizing* applications where it may not be very difficult to specify the approximate slope of the curve.
- ❖ These curves are preferred in the design of automotive, shipbuilding and aircraft industries, mechanical and structural components.

In computer graphics, the use of Hermite curves is *restricted* due to the following reasons:

- ➢ Quite cumbersome to select the *magnitude / angle of the tangent vectors* at the two endpoints of the curve segment.
- ➢ Hermite curves are cubic in nature; therefore, possesses $C^2$ continuity. Hence, it is difficult to control the curve due to the *global shape control* characteristics.
- ➢ The order of polynomial is always *cubic (constant)* irrespective of the number of control points.

## Limitations…

- A curve will be more flexible if a greater number of control points could be added, thus creating more curves which are all still of *cubic* order.
- If numbers of control points are large, the computation time required to invert the tangent vector matrix can be excessive.
- The cubic curves frequently exhibit spurious oscillations. The oscillations occur because each data point influences the cubic curve locally and third derivative is only piecewise constant.

**Example**

*The four control points in two-dimensional plane are* $P_0(0,0)$, $P_1(1,1)$, $P_2(2,-1)$ *and* $P_3(3,0)$. *The tangent vectors at the endpoints are* $P_0^{'}(1,1)$ *and* $P_3^{'}(1,1)$. *Determine the intermediate points on the Hermite curve at* $t=\frac{1}{3},\frac{2}{3}$.

**Solution:** A Hermite curve, in blending function formulation, is expressed as

$$P(t)=P_o.H_o(t)+P_3.H_1(t)+P_o^{'}.H_2(t)+P_3^{'}.H_3(t)$$

where Hermite blending functions are

$$H_o(t)=2t^3-3t^2+1\ ;\ H_1(t)=-2t^3+3t^2\ ;\ H_2(t)=t^3-2t^2+t\ ;\ H_3(t)=t^3-t^2$$

at $t=\frac{1}{3}$, $H_o(\frac{1}{3})=\frac{20}{27}$, $H_1(\frac{1}{3})=\frac{7}{27}$, $H_2(\frac{1}{3})=\frac{4}{27}$ and $H_3(\frac{1}{3})=\frac{-2}{27}$

at $t=\frac{2}{3}$, $H_o(\frac{2}{3})=\frac{7}{27}$, $H_1(\frac{2}{3})=\frac{20}{27}$, $H_2(\frac{2}{3})=\frac{2}{27}$ and $H_3(\frac{2}{3})=\frac{-4}{27}$

Hence, $P(\tfrac{1}{3}) = \begin{Bmatrix} x(\tfrac{1}{3}) \\ y(\tfrac{1}{3}) \end{Bmatrix} = \begin{Bmatrix} 0 \\ 0 \end{Bmatrix} . \left(\frac{20}{27}\right) + \begin{Bmatrix} 3 \\ 0 \end{Bmatrix} . \left(\frac{7}{27}\right) + \begin{Bmatrix} 1 \\ 1 \end{Bmatrix} . \left(\frac{4}{27}\right) + \begin{Bmatrix} 1 \\ 1 \end{Bmatrix} . \left(\frac{-2}{27}\right)$

$$= \begin{Bmatrix} 0 \\ 0 \end{Bmatrix} + \begin{Bmatrix} (7/9) \\ 0 \end{Bmatrix} + \begin{Bmatrix} (4/27) \\ (4/27) \end{Bmatrix} + \begin{Bmatrix} (-2/27) \\ (-2/27) \end{Bmatrix} = \begin{Bmatrix} (23/27) \\ (2/27) \end{Bmatrix}$$

and $P(\tfrac{2}{3}) = \begin{Bmatrix} x(\tfrac{2}{3}) \\ y(\tfrac{2}{3}) \end{Bmatrix} = \begin{Bmatrix} 0 \\ 0 \end{Bmatrix} . \left(\frac{7}{27}\right) + \begin{Bmatrix} 3 \\ 0 \end{Bmatrix} . \left(\frac{20}{27}\right) + \begin{Bmatrix} 1 \\ 1 \end{Bmatrix} . \left(\frac{2}{27}\right) + \begin{Bmatrix} 1 \\ 1 \end{Bmatrix} . \left(\frac{-4}{27}\right)$

$$= \begin{Bmatrix} 0 \\ 0 \end{Bmatrix} + \begin{Bmatrix} (20/9) \\ 0 \end{Bmatrix} + \begin{Bmatrix} (2/27) \\ (2/27) \end{Bmatrix} + \begin{Bmatrix} (-4/27) \\ (-4/27) \end{Bmatrix} = \begin{Bmatrix} (58/27) \\ (-2/27) \end{Bmatrix}$$

# COMPUTER AIDED DESIGN
## (BME-42)

### Unit-III: Space Curves
**(7 Lectures)**

- Properties for curve design, Parametric continuity,
- Parametric representation of synthetic curves, Spline curves and specifications, Parametric representation of synthetic curves
- Hermite curves-Blending functions formulation, shape control, properties,
- **Bezier curves-Blending functions formulation**, properties, Composite Bezier curves,
- Non-rational B-spline curves- Blending functions formulation, knot vector, B-spline blending functions, properties

# Lecture 22

### Topics Covered

**Bezier curves**

Nomenclature

Effects of Position of Control Points

Blending Function Formulation

*Prepared By*


Prof. S. K. SRIVASTAVA

MED, MMMUT, Gorakhpur (UP)
sksme@mmmut.ac.in

# BÉZIER CURVES

- Hermite curves are based on the *interpolation* techniques, i.e., curve passes through the existing data points.
- Bézier curves (named after a French Engineer *Pierre Etienne Bézier* for use in the design of Renault automobile bodies) are another alternative to generate curves based on the *approximation* techniques, produces the curves that do not pass through the given data (control) points.
- The outside data points control the shape of Bézier curves.
- In Bézier curve, we do not directly decide the magnitudes and angles of the tangent vectors; rather, we define the tangent vectors indirectly, by defining *two additional control points outside the curve.*
- In fact, two outside points control the shape of Bézier curves.

# BÉZIER CURVES…

Following are the major differences between the Bézier curve and Hermite curve:

- Its defining points control the shape of Bézier curve. This allows a much better feel for the relationship between the *input* (control points) and *output* (curve shape) parameters.
- Hermite curve is always **cubic** in nature. The degrees of polynomial are not related to the number of control points.
- The degrees of polynomial for the Bézier curve are variable and related to the number of control points defining the curve. For example, *$n^{th}$ degree curve requires $(n + 1)$ control points.*
- Higher degree Bézier curves permits *higher order continuity*.
- For Hermite curve, first derivatives are used for the curve development.

- Hermite curve possesses a *maximum of second order continuity* at the junction point of composite Hermite curves.

- *Bézier curve is smoother than Hermite curve* because it has higher order derivatives.

- Bézier curves are preferred for the ***ab initio*** design, i.e., design problems depending upon both aesthetic and functional requirements. For example, design of car bodies, aircraft fuselages, glassware, etc.

- Hermite curves are based on curve fitting techniques; therefore, not effective for the *ab initio* design problems.

## Nomenclature

Bézier curve is obtained by defining a characteristic polygon. Figure shows the nomenclature of a cubic (four control points) Bézier curve. The following observations can be made for the Bézier curves:

1. The curve is defined in terms of positions of the control points, the *vertices of Bézier characteristics polygon.*

2. The characteristics polygon uniquely defines the shape of the Bézier curve.

3. The Bézier curve *interpolates only the endpoints* and *approximates the remaining data points*.

4. The number of control points defining the curve determines the shape of Bézier curves.

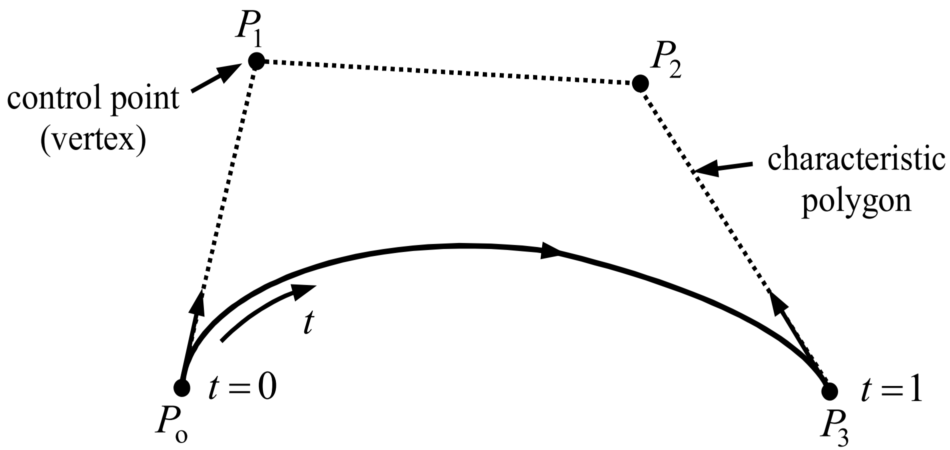
## Nomenclature…

5. The Bézier curve is *tangent to the first and last polygon segment* of the characteristic polygon.

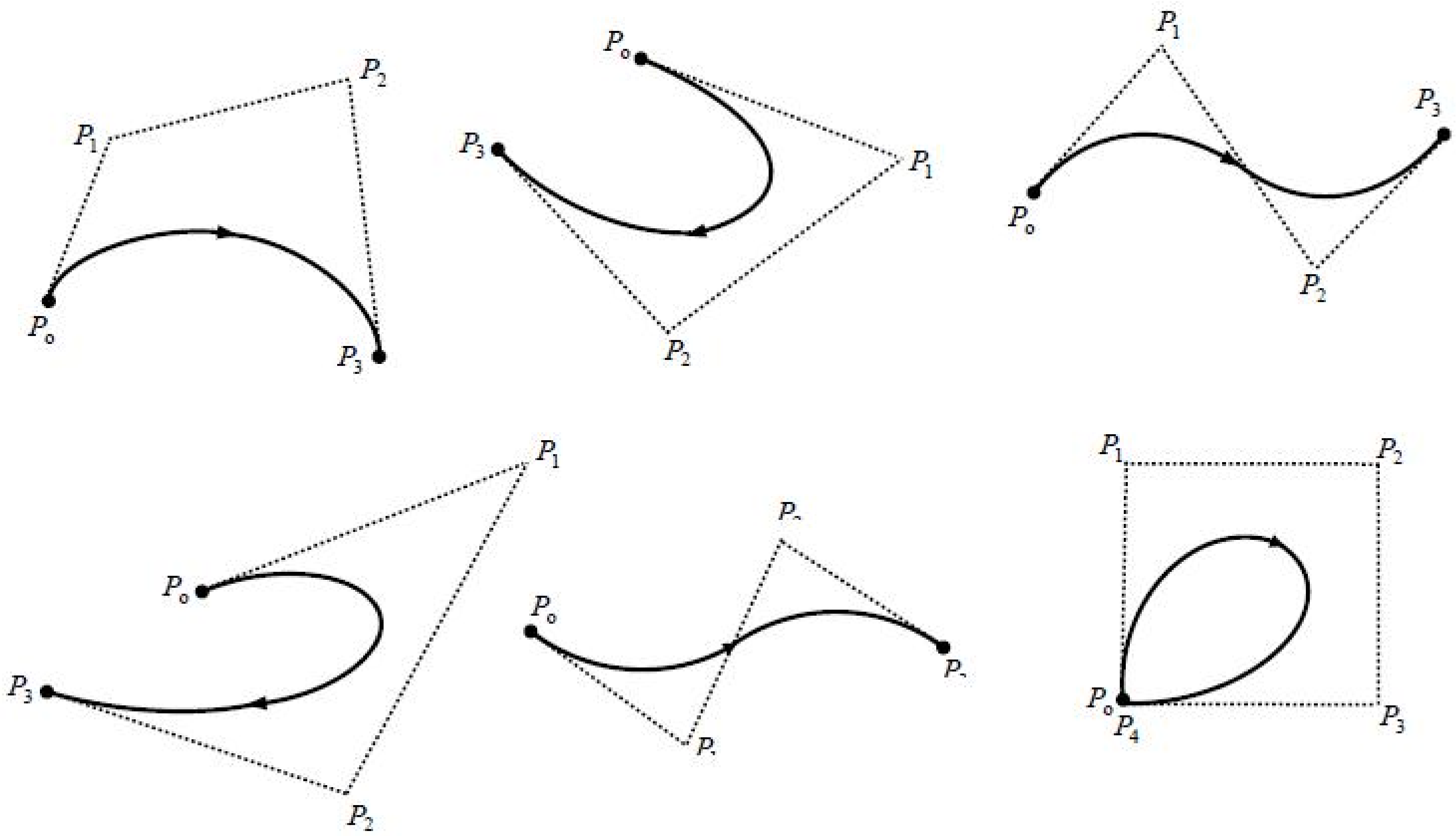
6. The Bézier curve *follows* the shape of *characteristic polygon*.

7. The arrow is directed from the parameter value $t = 0$ to $t = 1$. The direction of arrow shows the parametric direction of Bézier curve.

**Cubic Bézier curve**



## Effects of Position of Control Point

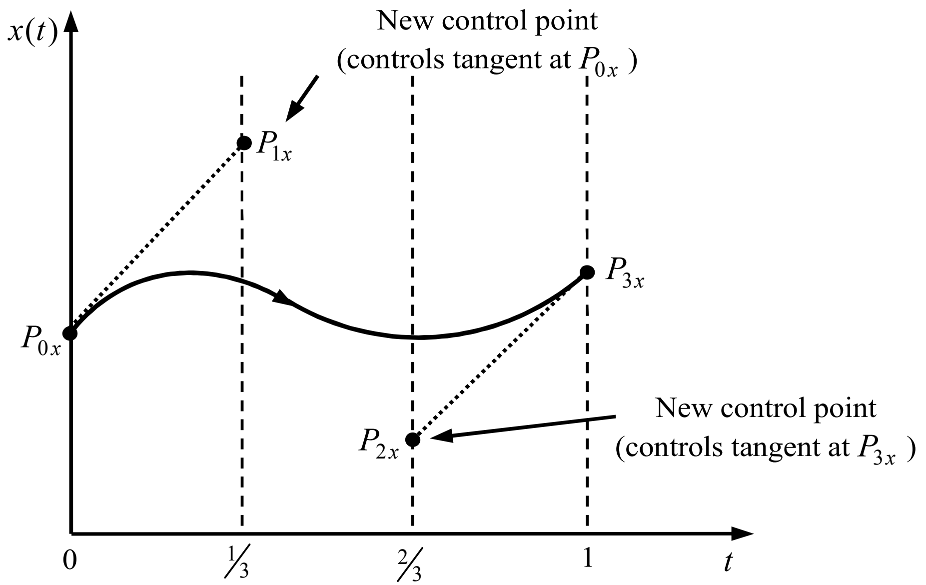



**Effect of position of control points on the shape of cubic Bézier curves**

## Blending Functions Formulation

The blending function formulation for defining the Bézier curve is most convenient. This requires the determination of *Bézier basis matrix* $M_B$ that characterizes the Bézier curve.

- Let us consider a plot for a cubic Bézier curve in *x*-direction only. The Bézier curve is like Hermite curve but the method of evaluation of cubic polynomial coefficients ,….. etc. is different. $a_x, b_x, c_x, d_x$
- In Bézier curves, the two additional control points $(P_{1x}$ and $P_{2x})$ outside the curve, control the tangents at the endpoints.




**The *x*(*t*) component of a cubic Bézier curve**

## Blending Functions Formulation…

- Thus, four control points (two endpoints $P_{ox}$ and $P_{3x}$, and two additional points outside the curve $P_{1x}$ and $P_{2x}$, are used to decide the shape of Bézier curve.

The tangent vectors $(P'_{ox}$ and $P'_{3x})$ at the endpoints of Bézier curve, in terms of the position of control points, is expressed as

at $t = 0$, $$P'_{ox} = x'(0) = \left.\frac{dx(t)}{dt}\right|_{t=0} = \frac{P_{1x} - P_{ox}}{\frac{1}{3} - 0} = 3(P_{1x} - P_{ox})$$

at $t = 1$, $$P'_{3x} = x'(1) = \left.\frac{dx(t)}{dt}\right|_{t=1} = \frac{P_{3x} - P_{2x}}{1 - \frac{2}{3}} = 3(P_{3x} - P_{2x})$$

The $x$ component of Bézier geometry matrix $(G_B)$ is defined as $$G_{Bx} = \begin{Bmatrix} P_o \\ P_1 \\ P_2 \\ P_3 \end{Bmatrix}_x$$

Let $M_{HB}$ defines a relation between Hermite geometry $(G_H)$ matrix and Bézier geometry matrix $(G_B)$ as

## Blending Functions Formulation…

$$G_{Hx} = M_{HB}.G_{Bx}$$

or $$G_{Hx} = \begin{Bmatrix} P_o \\ P_3 \\ P_o' \\ P_3' \end{Bmatrix}_x = \begin{Bmatrix} P_o \\ P_3 \\ 3(P_1 - P_o) \\ 3(P_3 - P_2) \end{Bmatrix}_x = \begin{Bmatrix} P_o \\ P_3 \\ -3P_o + 3P_1 \\ -3P_2 + 3P_3 \end{Bmatrix}_x = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ -3 & 3 & 0 & 0 \\ 0 & 0 & -3 & 3 \end{bmatrix} . \begin{Bmatrix} P_o \\ P_1 \\ P_2 \\ P_3 \end{Bmatrix}_x = M_{HB}.G_{Bx}$$

Thus, $$M_{HB} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ -3 & 3 & 0 & 0 \\ 0 & 0 & -3 & 3 \end{bmatrix}$$

For Hermite curve, we have $P(t) = T.M_H.G_H = T.M_H.(M_{HB}G_B) = T.M_B.G_B$

where Bézier basis matrix, $$M_B = M_H.M_{HB} = \begin{bmatrix} 2 & -2 & 1 & 1 \\ -3 & 3 & -2 & -1 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix} . \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ -3 & 3 & 0 & 0 \\ 0 & 0 & -3 & 3 \end{bmatrix}$$

as

## Blending Functions Formulation…

or

$$M_B = \begin{bmatrix} -1 & 3 & -3 & 1 \\ 3 & -6 & 3 & 0 \\ -3 & 3 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix}$$

Thus, parametric form of Bézier curve is expressed as

$$P(t) = T.M_B.G_B = \begin{bmatrix} t^3 & t^2 & t & 1 \end{bmatrix} \begin{bmatrix} -1 & 3 & -3 & 1 \\ 3 & -6 & 3 & 0 \\ -3 & 3 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix} . \begin{Bmatrix} P_o \\ P_1 \\ P_2 \\ P_3 \end{Bmatrix}$$

or $$P(t) = (-P_o + 3P_1 - 3P_2 + P_3).t^3 + (3P_o - 6P_1 + 3P_2).t^2 + (-3P_o + 3P_1).t + P_o$$

$$= P_o(1 - t^3 + 3t^2 - 3t) + P_1(3t^3 - 6t^2 + 3t) + P_2(-3t^3 + 3t^2) + P_3t^3$$

or $$P(t) = P_o(1-t)^3 + P_1.3t.(1-t)^2 + P_2.3t^2.(1-t) + P_3.t^3$$

## Blending Functions Formulation…

The polynomials, weights on each control points in the above equation, are termed *Bernstein polynomial* or *blending functions*. The Bernstein polynomial is defined as

$$B_{n,i}(t) = {}^{n}C_{i}\, t^{i}(1-t)^{n-i} \quad \text{when} \quad n \geq i$$

$$= 0 \quad \text{when} \quad n < i$$

where Bernstein polynomial $B_{n,i}(t)$ is known as $i^{th}$-$n^{th}$ order Bernstein basis function

And $${}^{n}C_{i} = \frac{n!}{i!(n-i)!}$$

where $n =$ degree of the defining Bernstein basis function, i.e., degree of polynomial of curve segment, one less than the number of control points, in the defining Bézier characteristic polygon

$n+1 =$ numbers of control points (vertices)

$i =$ particular control point (vertex) in the order (sequence)

## Blending Functions Formulation…

For example, for cubic Bézier curve, i.e., $n = 3$ ; the number of control points (vertices) are $n + 1$, i.e., 4. Thus, for $n + 1$ control points, the parametric Bézier curve of degree $n$ is defined as

$$P(t) = \sum_{i=0}^{n} P_i . B_{n,i}(t)$$

where $P(t)$ is any point on the Bézier curve and $P_i$ is a control point. The Bernstein polynomial works as blending or basis function for the Bézier curve. The values of Bernstein polynomial for $i = 0$ to $i = n$ may be obtained as

$$i = 0 \text{ and } t = 0, \quad B_{n,0}(0) = \frac{n!}{0!.(n-0)!}(0)^0.(1-0)^{n-0} = 1$$

$$i \neq 0 \text{ and } t = 0, \quad B_{n,i}(0) = \frac{n!}{i!.(n-i)!}(0)^i.(1-0)^{n-i} = 0$$

## Blending Functions Formulation…

$i = n$ and $t = 1,$ $\quad B_{n,n}(1) = \dfrac{n!}{0!..n!}(1)^{n}.(0)^{n-n} = 1$

$i \neq n$ and $t = 1,$ $\quad B_{n,i}(1) = \dfrac{n!}{i!.(n-i)!}(1)^{i}.(1-1)^{n-i} = 0$

where $(0)^0 \equiv 1$ and $0! \equiv 1$

Thus, above equation determines the parametric Bézier curve using Bernstein polynomial as blending function.

For *quadratic* polynomial ($n = 2$), the Bézier curve is expressed as

$$P(t) = \sum_{i=0}^{2} P_i.B_{n,i}(t) = P_o B_{2,0}(t) + P_1 B_{2,1}(t) + P_2 B_{2,2}(t)$$

or $$P(t) = P_o(1-t)^2 + P_1 2t(1-t) + P_2 t^2$$

## Blending Functions Formulation…

For *cubic* polynomial ($n$ = 3), the Bézier curve is expressed as

$$P(t) = \sum_{i=0}^{3} P_i . B_{n,i}(t) = P_o B_{3,0}(t) + P_1 B_{3,1}(t) + P_2 B_{3,2}(t) + P_3 B_{3,3}(t)$$

or $$P(t) = P_o (1-t)^3 + P_1 3t(1-t)^2 + P_2 3t^2 (1-t) + P_3 t^3$$

For *quartic* polynomial ($n$ = 4), the Bézier curve is expressed as

$$P(t) = \sum_{i=0}^{4} P_i . B_{n,i}(t) = P_o B_{4,0}(t) + P_1 B_{4,1}(t) + P_2 B_{4,2}(t) + P_3 B_{4,3}(t) + P_4 B_{4.4}(t)$$

or $$P(t) = P_o (1-t)^4 + P_1 4t(1-t)^3 + P_2 6t^2 (1-t)^2 + P_3 4t^3 (1-t) + P_4 t^4$$

## Blending Functions Formulation…

Figure a, b & c, respectively, show the plots for the Bézier/Bernstein blending functions for three, four and five control points.

For $0 \leq t \leq 1$ the following observations can be made from eqns.

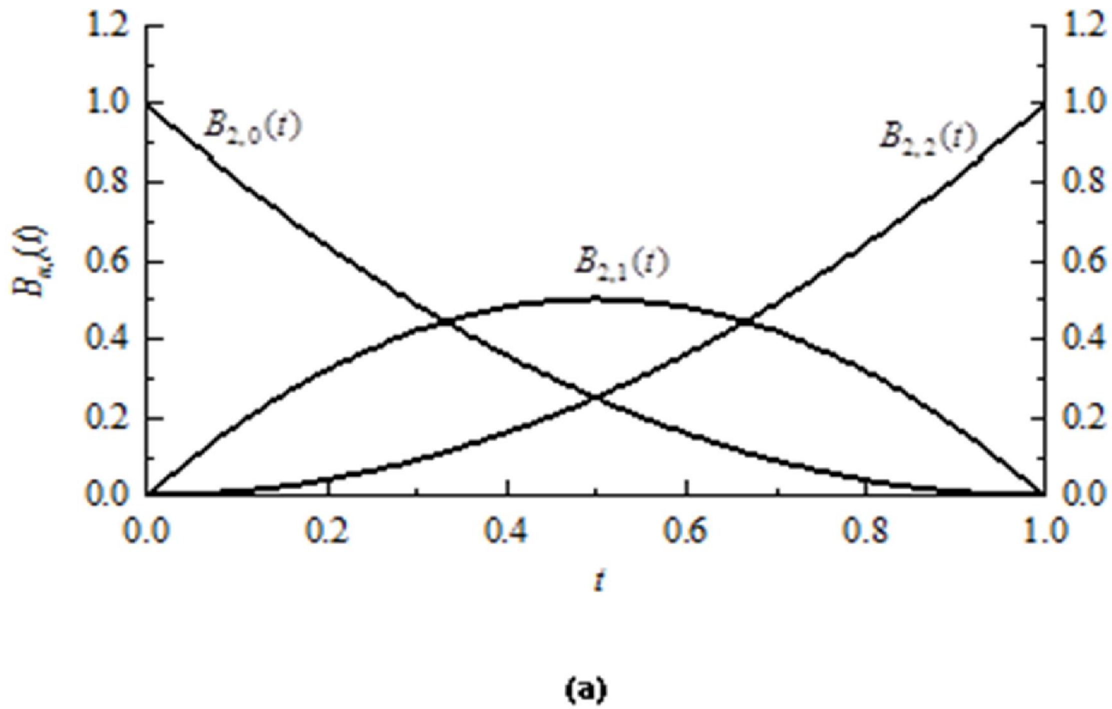
1. The sum of Bernstein polynomials is *unity* everywhere
2. Every polynomial is *non-negative* everywhere

Therefore, the position of a point on the Bézier curve, defined by $P(t)$, is just the weighted average of control points defining the curve.

Mathematically, for any value of parameter $t$ $(0 \leq t \leq 1)$, the summation of Bernstein basis functions is precisely equal to unity, i.e.

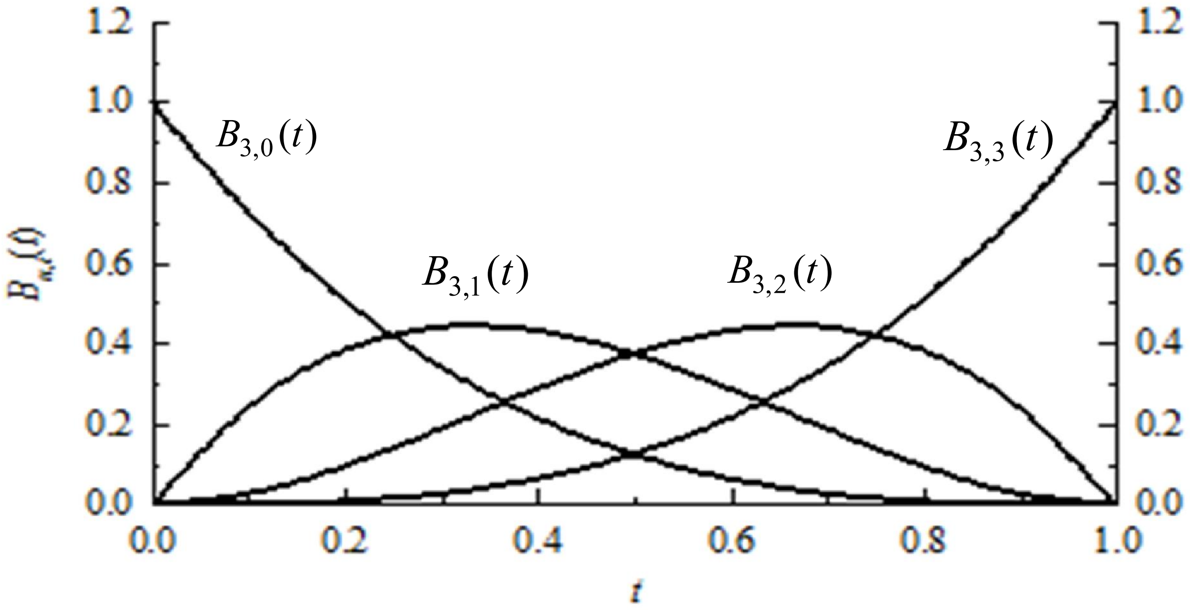
$$\sum_{i=0}^{n} B_{n,i}(t) = 1$$

Blending Functions Formulation…



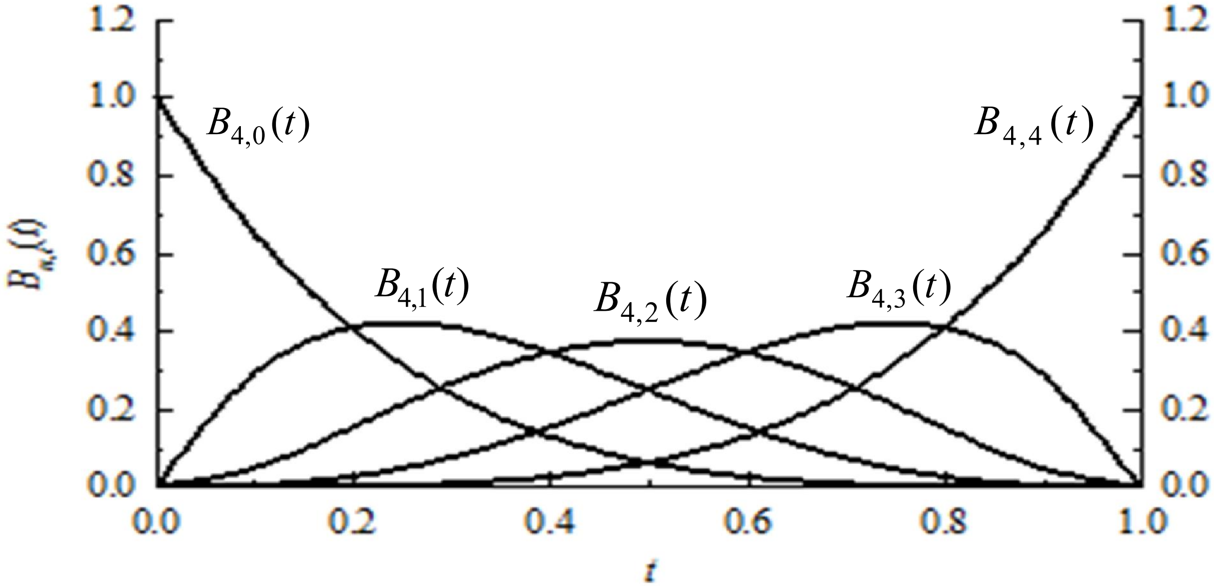

**(a) Bézier/Bernstein blending functions for three control points**

Blending Functions Formulation…




**(b) Bézier/Bernstein blending functions for four control points**

Blending Functions Formulation…




**(c) Bézier/Bernstein blending functions for five control points**

# COMPUTER AIDED DESIGN
## (BME-42)

### Unit-III: Space Curves

**(7 Lectures)**

- Properties for curve design, Parametric continuity,
- Parametric representation of synthetic curves, Spline curves and specifications, Parametric representation of synthetic curves
- Hermite curves-Blending functions formulation, shape control, properties,
- Bezier curves-Blending functions formulation, **properties, Composite Bezier curves**
- Non-rational B-spline curves- Blending functions formulation, knot vector, B-spline blending functions, properties

# Lecture 23

## Topics Covered

**Properties of Bezier Curves**

**Composite Bezier Curve**

**Drawbacks of Composite Bézier Curves**

**Essential Requirements for Synthetic Curves**


*Prepared By*

**Prof. S. K. SRIVASTAVA**

MED, MMMUT, Gorakhpur (UP)
sksme@mmmut.ac.in

# BÉZIER CURVES…

- In curve fitting techniques (Hermite curves), the *designer does not have proper control* over the shapes of the curve because they are **uniquely** defined within a specified interval.
- It is difficult to set up the *magnitudes* and *directions* of **tangent vector** at the endpoints.
- The Bézier curves are *interpolating* (passes through the endpoints) and *approximating* (approximates outside control points) polynomials.
- In Bézier curves, the designer has *sufficient control* over the shape of the curve. That is why these curves are preferred for the ***aesthetic*** design of a component where styling is required.
- The designer can set up the desired style/shape simply by *controlling positions* of the control points outside the Bézier curve.
- The properties of Bézier curves depend upon the properties of *Bernstein polynomials*.

# PROPERTIES OF BÉZIER CURVES...

The properties of Bézier curve are

- The curve *interpolates* the first and last control points, i.e., it passes through $P_o$ and $P_n$ control points corresponding to the parameter $t = 0$ and $t = 1$, respectively.

- Bézier curve is *tangent to the first and last segments* of the characteristic polygon; therefore, it maintains tangent vectors continuity $(C^1)$ at the endpoints when joined with other segment of the Bézier curve.

- The Bernstein blending functions are *real*.

- The curve generally follows the *shape of the characteristic polygon*. The curve is tangent to the first and last segments of the characteristic polygon.

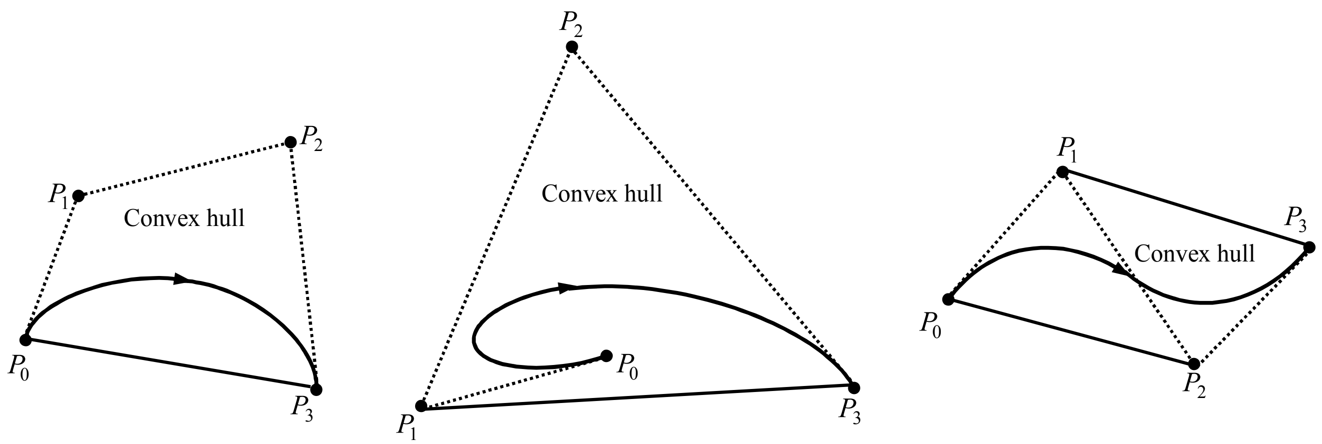
- The number of control points specified within the curve segment defines the degree of a Bézier curve. For $(n + 1)$ data points within the specified curve segment, the degree of polynomial defining the Bézier curve will be $n$.

# PROPERTIES OF BÉZIER CURVES…

- The shape (hence, *degree*) of a Bézier curve can be modified by adding or deleting the control points. This is the most useful property usually desired by the designer.
- Compared to Hermite curves, the *blending functions* for the Bézier curve are all positive and their sum is always equal to unity.
- Due to this property, the curve lies within the convex of the defining polygon, i.e., it remains within the *convex hull*.
  - In 2D, the convex hull is a closed polygon.
  - It can be considered a rubber band stretched around the positions of all the control points so that each control point is either on the perimeter of the hull or inside it.
  - In 3D, the convex hull is a balloon (polyhedron) touching all the control points in space. The region inside the balloon is convex hull.

# PROPERTIES OF BÉZIER CURVES…

- The volume of the region changes with the positions of control points.
- Size of the convex hull provides an upper bound on the size of Bézier curve itself, i.e., the curve lies within the convex hull.
- Convex hulls provide a measure for the deviation of two-dimensional curves or three-dimensional surfaces from the region bounded by the control points.
- The convex hull ensures the smoothness of the curves/surfaces following the control points without any deviation or oscillations.
- Figure shows the convex hulls shown by the closed polygons.



# PROPERTIES OF BÉZIER CURVES...

- The Bézier curve exhibits the *variation diminishing property* because of the convex hull property.
  - This means that the curve never oscillates widely away from the defining control points of the characteristic polygon because the curve is guaranteed to lie within the convex hull.
  - Alternatively, the curve does not oscillate about any straight line (generally the sides of the characteristic polygon) more often than the sides of its defining polygon.
- The Bézier curve is invariant under an affine transformation.
  - An affine transformation is a combination of linear transformations, e.g., rotation followed by the translation.
  - For an affine transformation, the last row in a general 4x4 transformation matrix is [0 0 0 1].

# PROPERTIES OF BÉZIER CURVES…

- Any affine transformation can be applied to the curve, by applying it to the defining polygon vertices, i.e., the curve is transformed by transforming vertices of the characteristic polygon.

➢ Parametric transformation of Bézier curve is possible. If transformation parameter is changed from $0 \le t \le 1$ to $a \le t \le b$, then parameter $t$ should be replaced by $(t-a)/(b-a)$

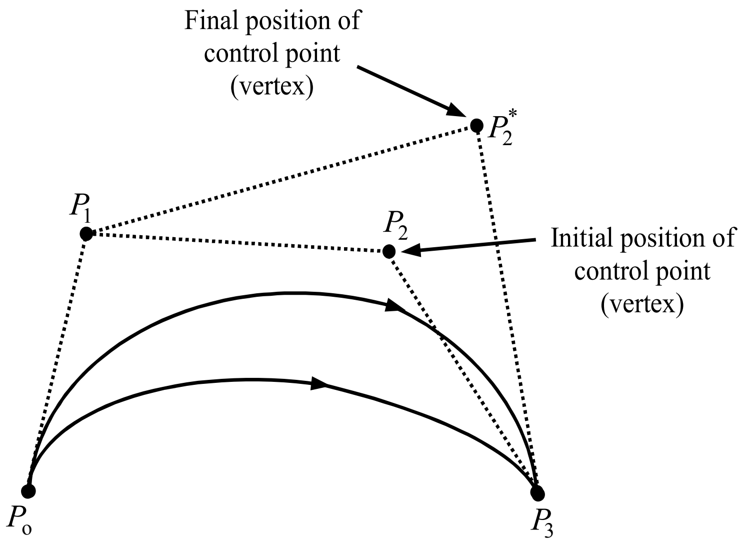
➢ The Bézier curve is *symmetric* with respect to the parameter $t$ and $(1 - t)$. Thus, the shape of the curve remains same if sequence of control points, defining the curve, is reversed.

➢ A closed Bézier curve can be generated by making first and last control points to coincide.

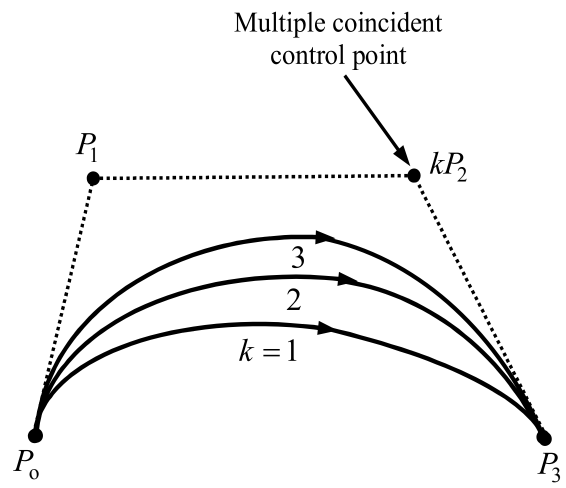
➢ For any degree of Bézier curve, the sum of Bernstein blending functions associated with the control points is always equal to unity, for any value of parameter $t$ (i.e., $0 \le t \le 1$. This property checks the numerical computations during the software development.

# PROPERTIES OF BÉZIER CURVES…

- *Bézier curves do not provide local control*, i.e., movement of any control point changes the entire shape of the curve. This is because of the property of Bernstein functions.
- The shape of Bézier curve modifies by changing the position of one or more vertices of the characteristic polygon. Figure (a) depicts change in the shape of Bézier curve when vertex $P_2$ is pulled to the new location $P_2^*$.




(a)




(b)

# PROPERTIES OF BÉZIER CURVES…

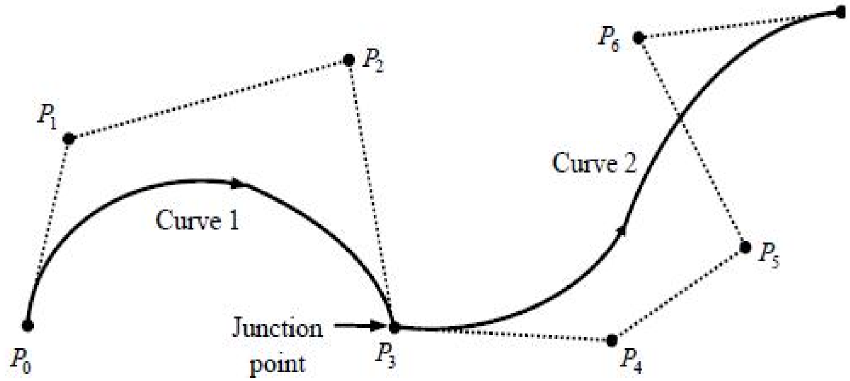
- The shape of Bézier curve can be modified by specifying the *multiple coincident control points at a vertex*, keeping the characteristic polygon fixed.
- Figure (b) depicts change in the shape of Bézier curve when the vertex $P_2$ is assigned a multiplicity of $k$. Higher the multiplicity, more the curve pulled towards the control point $P_2$ .

## Composite Bézier curves

- Many CAD applications require the composite Bézier curves in which various curve segments are joined to generate a longer curve.
- They require maintaining the *continuity of various orders* between the curve segments.
- Figure shows the two curve segments defined by the two sets of control points, i.e. $P_o$, $P_1, P_2, P_3$ and $P_3, P_4, P_5, P_6, P_7$ joined at $P_3$
- Four control points result into a cubic Bézier curve whereas five control points define a Bézier curve comprising of fourth degree polynomial.

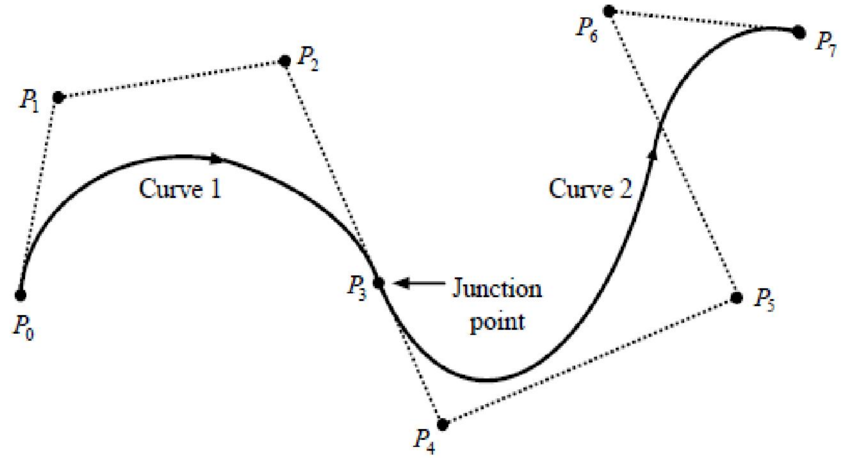
# COMPOSITE BÉZIER CURVES

Zero Order Continuity or $C^0$ Continuity




Position continuity $(C^o)$ exists when one of the end control point $(P_3)$ of the two curve segments is common

First Order Continuity or $C^1$ Continuity




The tangent continuity $(C^1)$ at the common (junction) point $(P_3)$ exists if end slope of first curve segment is equal to the starting slope of the second curve segment.

Common point and its two adjacent points must be collinear

For $C^2$ Continuity, the common point and its four adjacent points must be collinear

# COMPOSITE BÉZIER CURVES…

- Alternatively, when tangent vectors for the two curve segments at the common (junction) point relates each other by a constant then $C^1$ continuity exists for the composite Bézier curves.
- In other words, $C^1$ continuity at the common point requires that the last segment of the first characteristic polygon and first segment of the second characteristic polygon are collinear (forms a straight line). Thus, control points $P_2$, $P_3$, $P_4$ are collinear.

The tangent vectors at the endpoints of $n^{th}$ degree Bézier curve, defined by the ($n$ + 1) control points, is given as

$$P'(0) = n(P_1 - P_o) \text{ at } t = 0$$

$$P'(1) = n(P_n - P_{n-1}) \text{ at } t = 1$$

# COMPOSITE BÉZIER CURVES…

For $C^1$ continuity at the common point, the collinearity requires that

$$\begin{pmatrix} \textit{Tangent vector at the last control point} \\ \textit{of the first curve segement} \end{pmatrix} = \begin{pmatrix} \textit{Tangent vector at the first control point} \\ \textit{of the econd curve segement} \end{pmatrix}$$

Here, for the first curve segment $P'(1) = 3(P_3 - P_2)$

And, for the second curve segment $P'(0) = 4(P_4 - P_3)$

Thus, $3(P_3 - P_2) = 4(P_4 - P_3)$

or $P_3 - P_2 = \frac{4}{3}(P_4 - P_3)$

Therefore, tangent vectors at the common control point $(P_3)$ for the two Bézier curves segments, defined by $P_o, P_1, P_2, P_3$ and $P_3, P_4, P_5, P_6, P_7$ control points, are related to each other by a constant equal to 4/3.

**Example:** *Draw a Bézier curve defined by the four control points* $P_0(1,2)$ , $P_1(3,4)$ , $P_2(6,-6)$ *and* $P_3(9,7)$ .

**Solution:** The parametric form of Bézier curve, defined by the four control points, is given as

$$P(t)=\sum_{i=0}^{3}P_i.B_{n,i}(t)=P_oB_{3,0}(t)+P_1B_{3,1}(t)+P_2B_{3,2}(t)+P_3B_{3,3}(t)$$

or $$P(t)=P_o(1-t)^3+P_1 3t(1-t)^2+P_2 3t^2(1-t)+P_3t^3$$

Therefore, any point $P(t)$ on Bézier curve is given by

$$P(t)=\begin{Bmatrix}x(t)\\y(t)\end{Bmatrix}=\begin{Bmatrix}1\\2\end{Bmatrix}.(1-t)^3+\begin{Bmatrix}3\\4\end{Bmatrix}.3t(1-t)^2+\begin{Bmatrix}6\\-6\end{Bmatrix}.3t^2(1-t)+\begin{Bmatrix}9\\7\end{Bmatrix}.t^3$$

On simplification, $x(t)$ and $y(t)$ coordinates are given as

$$x(t)=(1-t)^3+9t(1-t)^2+18t^2(1-t)+9t^3$$

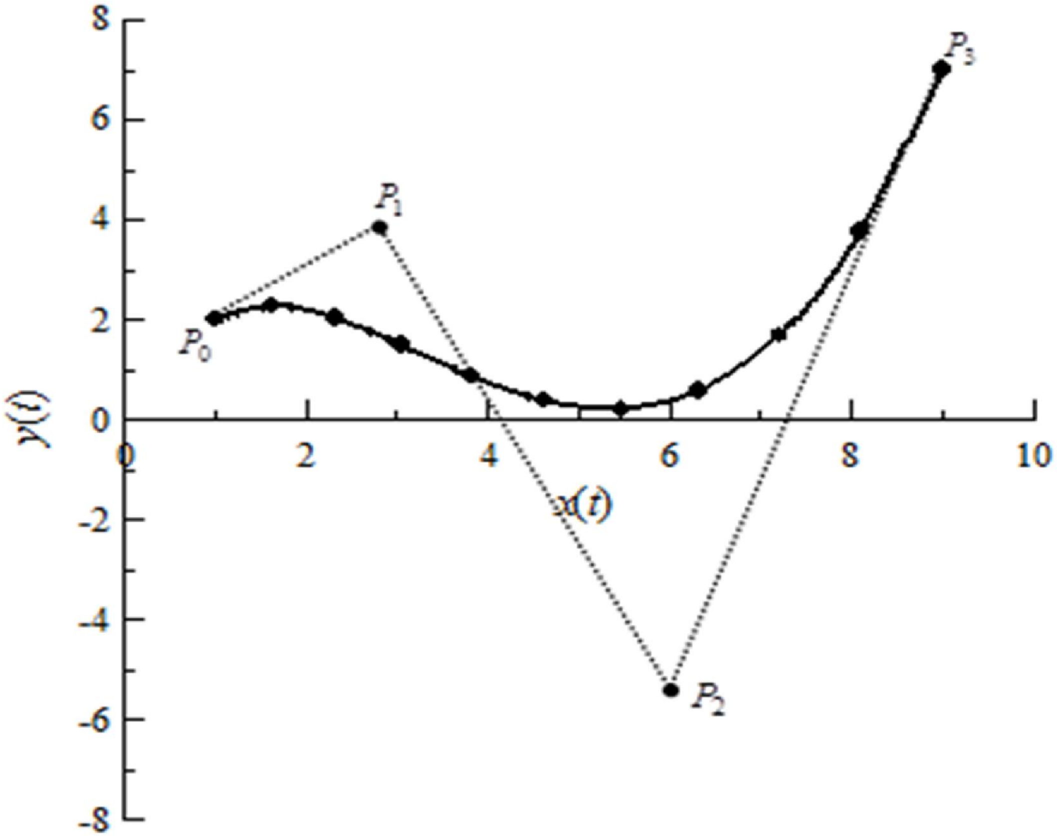
$$y(t)=2(1-t)^3+12t(1-t)^2-18t^2(1-t)+7t^3$$

For step size $t = 0.1$, the corresponding values of $x\ (t)$ and $y\ (t)$ coordinates are tabulated in Table. Figure shows the plot for Bézier curve segment.

***Table*** *Calculation of* $x(t)$ *and* $y(t)$ *coordinates for step size* $t = 0.1$

| $t$ | $x(t)$ | $y(t)$ | $t$ | $x(t)$ | $y(t)$ |
|---|---|---|---|---|---|
| 0.0 | 1.000 | 2.000 | 0.6 | 5.464 | 0.200 |
| .1 | 1.629 | 2.275 | 0.7 | 6.327 | 0.565 |
| 0.2 | 2.312 | 2.040 | 0.8 | 7.208 | 1.680 |
| 0.3 | 3.043 | 1.505 | 0.9 | 8.101 | 3.755 |
| 0.4 | 3.816 | 0.880 | 1.0 | 9.000 | 7.000 |
| 0.5 | 4.625 | 0.375 | - | - | - |

# BÉZIER CURVES…



# DRAWBACKS OF COMPOSITE BÉZIER CURVES…

A designer faces the following problems while using the composite Bézier curve segments:

- With Bézier curves, only control (data) points are specified. The curve segments have *no local control* (local change affects the entire shape of the curve) due to the properties of Bernstein (blending) functions. Therefore, the designer cannot selectively change parts of the curve.

- The Bézier curve does not interpolate the control points (except the endpoints), which may be inconvenient to the designer. Interpolation is useful in design or engineering results such as displaying the stress distribution in a component obtained from finite element analysis.

- Composite Bézier curves impose constraints on the location of control points. For example, slope continuity $(C^1)$ at the common point requires that the common point and its two adjacent control points on either side of the curve segment must be *collinear*.

# DRAWBACKS OF COMPOSITE BÉZIER CURVES…

- $C^2$ continuity (slope derivative) at the common point of composite Bézier curves further extends the constraint on the location of control points.

- It requires four control points, in addition to the common point, to lie on a plane; therefore, restrict the freedom of choosing the data points for the composite Bézier curves.

- Thus, it is difficult to achieve $C^2$ continuity for the composite Bézier curves, with lower degree polynomials.

- Keeping this in view, the designer prefers Bézier curve segments with an order 6 or 8 (hence, degree 5 or 7, respectively) for most of the CAD applications.

# DRAWBACKS OF COMPOSITE BÉZIER CURVES…

- The number of specified polygon vertices fixes the order (hence, degree) of the resulting polynomial defining the Bézier curve.

- For example, a cubic curve by the four vertices and three spans.
- A fifth degree Bézier curve requires six vertices in the characteristic polygon.
- Thus, degrees of polynomial have been liked with the number of vertices.
- Therefore, to reduce degrees of the curve (also reduces the computation time) is to reduce the number of vertices in the polygon.
- Alternatively, the degree of curve can only be increased, by increasing the number of control points.

# ESSENTIAL REQUIREMENTS FOR SYNTHETIC CURVES

In computer graphics, the designer can use a curve in a more comfortable way if the following facilities are available:

1. **Local modification over any segment of the curve**

   The designer should be able to change the positions of the control points in an intuitive way without changing the overall shape (global change) of the entire curve segment.

2. **Delink the number of control points and the degree of polynomial**

   The designer should be able to use lower degree polynomial segments still maintaining the shape of curve using large number of control points.

# ESSENTIAL REQUIREMENTS FOR SYNTHETIC CURVES

**3. Parametric piecewise curve fitting with $C^2$ continuity**

This is desirable for a curve to be inherently $C^2$ continuous throughout the length.

**4. Finer shape control of curve by the knots insertions**

Knots provide additional tool for designing and local editing of the curve shape.

# COMPUTER AIDED DESIGN
## (BME-42)

### Unit-III: Space Curves

**(7 Lectures)**

- Properties for curve design, Parametric continuity,
- Parametric representation of synthetic curves, Spline curves and specifications, Parametric representation of synthetic curves
- Hermite curves-Blending functions formulation, shape control, properties,
- Bezier curves-Blending functions formulation, properties, Composite Bezier curves,
- **Non-rational B-spline curves-Blending functions formulation, knot vector**, B-spline blending functions, properties

# Lecture 24

## Topics Covered

B-Spline Curves

Characteristics

Advantages over the Bezier Curve

Types of B-Spline Curves

Non-rational B-Spline Curves

B-Spline Blending Functions

B-spline Blending Function Formulation

Dependency Diagram for Blending Function

*Prepared By*

Prof. S. K. SRIVASTAVA

MED, MMMUT, Gorakhpur (UP)

sksme@mmmut.ac.in

# B-SPLINE CURVES

- Piecewise joining of polynomials gives splines.
- The letter 'B' stands for the *basis*, since the splines are represented as weighted sums of the polynomial basis (blending) functions, in contrast to the natural splines.

**Those polynomials (or spline blending functions) that gives minimum span or maximum possible control over the curve are termed B-Splines**

## Characteristics

- B-spline curves are characterized by approximating the endpoints, allowing first and second order derivatives to be continuous at the endpoints of the curve.
- *Under **certain** conditions, the curve may interpolate the endpoints.*

Consider blending functions $R_k(t)$ designated by blending function like Bernstein polynomial in a Bézier curve. It is some *hypothetical* blending function having the variation within the range $0 \le t \le 1$

A point defined by the hypothetical blending functions may be expressed as

$$P(t) = \sum_{k=0}^{n} P_k . R_k(t) \qquad 0 \le t \le 1$$

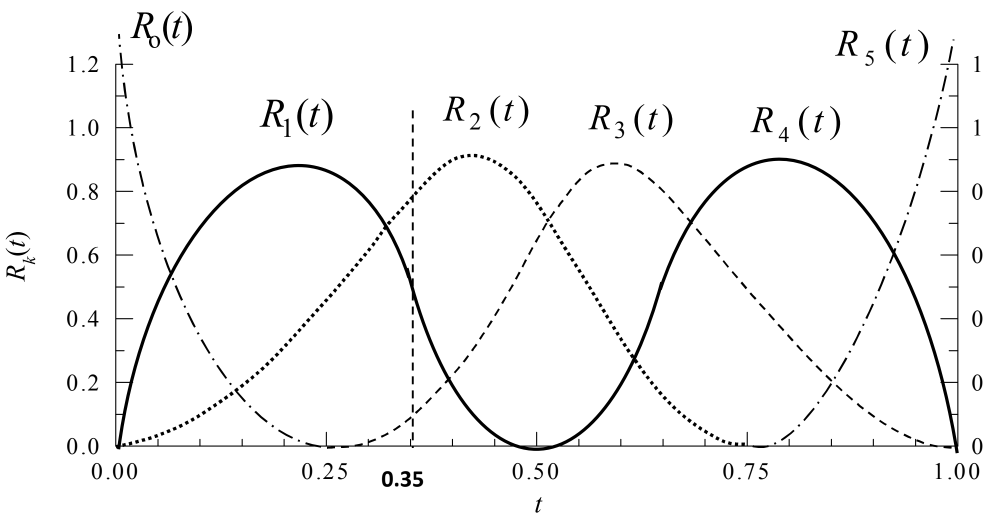
For n + 1 = 6 vertices or n = 5 degrees of polynomial of blending functions, the points on the hypothetical curve is given as

$$P(t) = \sum_{k=0}^{5} P_k . R_k(t) = P_o R_o + P_1 R_1(t) + P_2 R_2(t) + P_3 R_3(t) + P_4 R_4(t) + P_5 R_5(t)$$

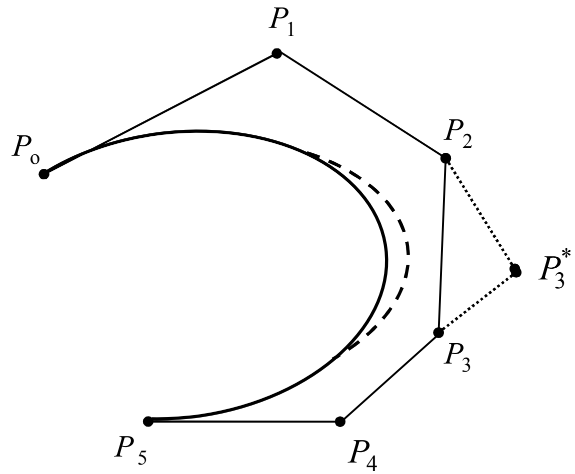
Figure show the variations of hypothetical blending functions $R_k(t)$ and the corresponding curve. It can be observed that

$$\text{at } t = 0 \qquad P(t) = P_o R_o(t)$$

$$\text{at } t = 1 \qquad P(t) = P_5 R_5(t)$$




Hypothetical blending functions




Spline depicting local control

In contrast to Bernstein blending functions (nonzero for $0 \le t \le 1$), the hypothetical blending functions, $R_i(t)$, are non-zero only for small intervals. we observed that

$R_o(t)$ is non-zero only for the span from $t = 0 - 0.25$,

$R_1(t)$ is non-zero only for the span from $t = 0 - 0.50$,

$R_2(t)$ is non-zero only for the span from $t = 0 - 0.75$,

$R_3(t)$ is non-zero only for the span from $t = 0.25 - 1.0$,

$R_4(t)$ is non-zero only for the span from $t = 0.50 - 1.0$, and

$R_5(t)$ is non-zero only for the span from $t = 0.75 - 1.0$.

- *Hence, if range of the blending functions is less than t = 0 – 1, we have better control over the shape of the curve.*
- Alternatively, the curve behave like *non-global*, i.e., portion of the curve can be modified without changing the overall shape of the entire curve.

*Polynomials* are preferred as a blending function due to following reasons:

- *Easier to control*
- *Easier to check for the continuity*

  (*easy differentiation compared to other types of functions*)

- B-spline curves have ability to *interpolate* or *approximate* the given set of control points.
- In many engineering applications, interpolation is useful, e.g., displacements or stress distribution in a component under the load.
- Interpolation is also useful if designer has measured the data points, and a curve passing through these data points is generated.
- Similarly, *approximation is used for drawing the free-form curves*.

## Advantages over the Bézier curves

- The degree of B-spline polynomial can be set independently of number of control points.
  - The B-spline curves *delink* the degrees of resulting curves from the number of control points.
  - For example, *four control points always result into a cubic Bézier curve*; but four control points in B-spline curve can generate a *linear, quadratic or cubic* curves.
  - This flexibility in B-spline curves is obtained by *choosing the blending functions with additional degree of freedom*, which is not available with Bernstein blending functions

## Advantages over the Bézier curves…

➢B-spline functions allow local control over the shape of the curve.

- The polynomial coefficient depends on just a few control points, leading to *local control* over the shape of the curve.

- Bernstein blending functions is a special case of B-spline blending functions.

- B-spline curves have $C^2$ continuity, like natural splines, but *do not interpolate their control points.*

,

# TYPES OF B-SPLINE CURVES…

1. Non- Rational B-Splines
   a) *Periodic Uniform Knot Vector*
   b) *Open Uniform Knot Vector*
   c) *Non -Uniform Knot Vector*

2. Rational B-Splines

   Projection of non-rational defined in 4D homogeneous coordinate space (often called *weights on blending functions*) into 3D physical space

   a) *Periodic Uniform Knot Vector*
   b) *Open Uniform Knot Vector*
   c) *Non-Uniform Knot Vector*

## B-Spline Blending Functions

*Splines are used as blending functions.*

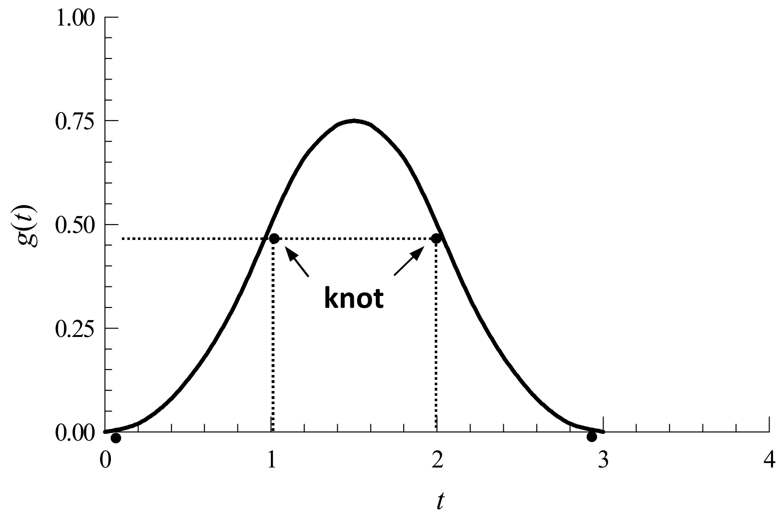
Let, the *quadratic* ($m = 2$) spline function is defined as

$$g(t) = \begin{cases} a(t) = \frac{1}{2}t^2 & 0 \le t \le 1 \\ b(t) = \frac{3}{4} - (t - \frac{3}{2})^2 & 1 \le t \le 2 \\ c(t) = \frac{1}{2}(3-t)^2 & 2 \le t \le 3 \end{cases}$$

Knots are provided at $t = 0, 1, 2, 3$

$a(1) = b(1)$ $a'(1) = b'(1)$ and $a''(1) \neq b''(1)$

$b(2) = c(2)$ $b'(2) = c'(2)$ and $b''(2) \neq c''(2)$



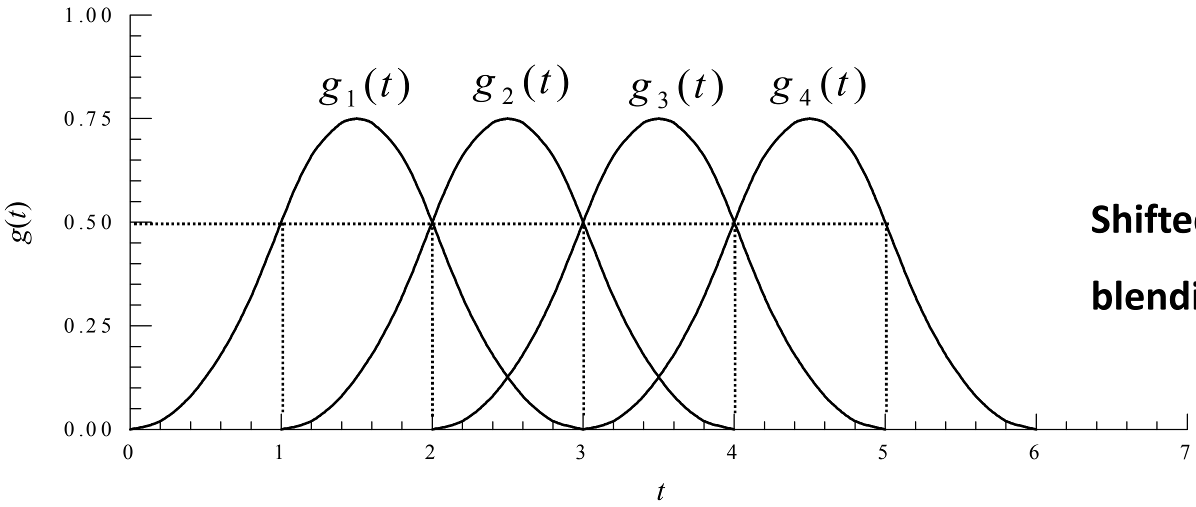

- $m^{th}$ degree spline is a piecewise polynomial of $m^{th}$ degree,
- which has its first *(m-1)* derivatives continuous.

# NON-RATIONAL B-SPLINE CURVES…

- The spline functions are used as blending functions in B-splines.
- The shifted splines can be used as blending functions, i.e., other spline blending functions can be written as

$$g_i(t) = g(t-i)$$

Each blending functions are simply translate of the others. The number of spline blending functions depends upon the number of control points. For example, the *quadratic* blending functions corresponding to the *four* control points are expressed as




**Shifted quadratic $(m=2)$ spline blending functions**

# NON-RATIONAL B-SPLINE CURVES…

for $i=1$ — $g(t)$ or $g_1(t)$ are same

for $i=2$ — $g_2(t)=g(t-1)$, $g_2(t)$ is same as $g(t)$ or $g_1(t)$ but shifted at $t=1$

for $i=3$ — $g_3(t)=g(t-2)$, $g_3(t)$ is same as $g(t)$ or $g_1(t)$ but shifted at $t=2$

for $i=4$ — $g_4(t)=g(t-3)$, $g_4(t)$ is same as $g(t)$ or $g_1(t)$ but shifted at

- Quadratic splines can be used as blending functions for the generation of 2D B-splines.
- To generate the curve, at least *four* control points are required, and each blending function will weight one control point.
- The B-spline curve can be expressed as

$$P(t) = \sum_{i=1}^{n+1} P_i g_i(t) \qquad t_{\min} \le t \le t_{\max}$$

where $i$ represents a knot or control point.

# NON-RATIONAL B-SPLINE CURVES…

For quadratic spline blending functions ($m = 2$) with four control points (knots), i.e., $n + 1 = 4$ or $n = 3$, the B-spline function is expressed as

$$P(t) = \sum_{i=1}^{n+1} P_i g_i(t) = P_1 g_1(t) + P_2 g_2(t) + P_3 g_3(t) + P_4 g_4(t) \qquad 0 \le t \le n+3, i.e. 6$$

From Figure, the different points $P(t)$ on B-spline curve may be calculated as

at $t = 0$, $g_1(t) = g_2(t) = g_3(t) = g_4(t) = 0$; $P(t) = 0$ is the first point (origin itself)

at $t = 1$, $g_1(t) = \frac{1}{2}$ and $g_2(t) = g_3(t) = g_4(t) = 0$; $P(t) = \frac{P_1}{2}$

at $t = 2$, $g_1(t) = g_2(t) = \frac{1}{2}$ and $g_3(t) = g_4(t) = 0$; $P(t) = \frac{P_1 + P_2}{2}$

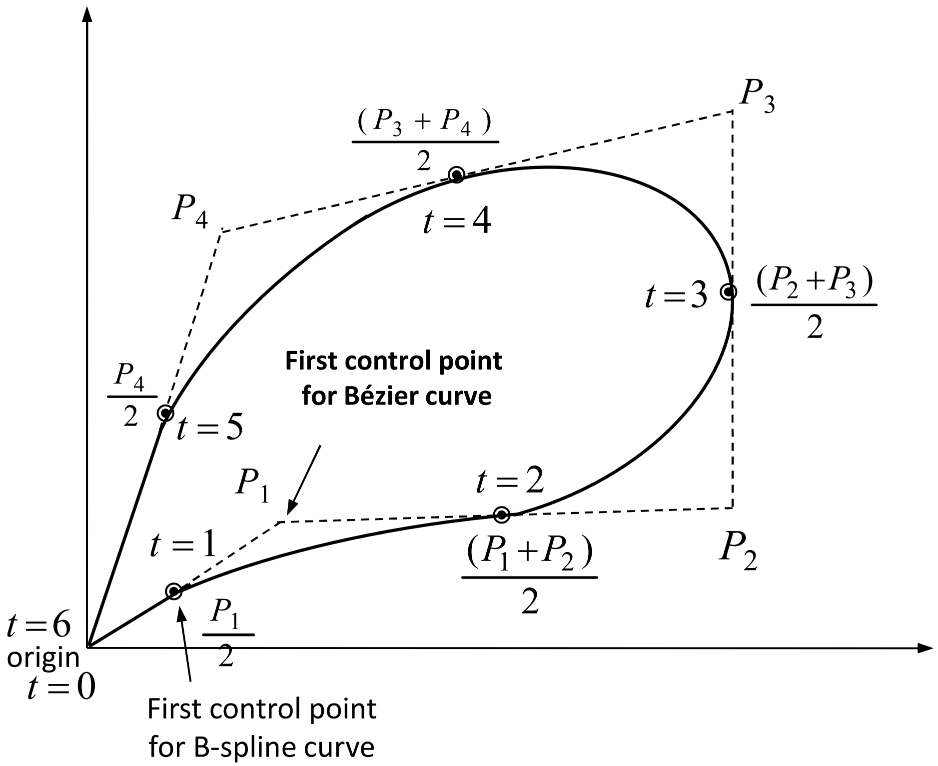
at $t = 3$, $g_2(t) = g_3(t) = \frac{1}{2}$ and $g_1(t) = g_4(t) = 0$; $P(t) = \frac{P_2 + P_3}{2}$

at $t = 4$, $g_3(t) = g_4(t) = \frac{1}{2}$ and $g_1(t) = g_2(t) = 0$; $P(t) = \frac{P_3 + P_4}{2}$

at $t = 5$, $g_4(t) = \frac{1}{2}$ and $g_1(t) = g_2(t) = g_3(t) = 0$; $P(t) = \frac{P_4}{2}$

at $t = 6$, $g_1(t) = g_2(t) = g_3(t) = g_4(t) = 0$; $P(t) = 0$ is the last point (origin itself)

Figure shows B-spline curve generated by joining the above points. In general, parameter $t$ ranges from $t = 0$ to $t = n + 3$, i.e., 6




**B-spline curve with quadratic shifted splines as blending functions**

# NON-RATIONAL B-SPLINE CURVES…

- The drawback of B-spline curve is that it is *not passing through the endpoints*, i.e., endpoints $P_1$ and $P_4$, as obtained in Bézier curve.
- In Bézier curve, the designer can fix the two extreme values of parameter $t$ for the endpoints, and curve passes through the endpoints; therefore, easy to control the curve shape.
- However, in B-spline curves, the first endpoint occurs at the middle point of line joining the first endpoint with the origin.
- Similarly, the last endpoint occurs at the middle point of line joining the last endpoint with the origin.
- To make the curve to pass through the endpoints (say, first endpoint $P_1$), the designer can select the first endpoint $P_1$ so that $\frac{P_1+P_2}{2}$ falls at the endpoint $P_1$. This is obtained by using the *multiplicity of control points*.

# NON-RATIONAL B-SPLINE CURVES…

- Moreover, the *origin can be avoided* if parameter ***t*** ranges from $2 \le t \le n+1$ , instead of $0 \le t \le n+1$
- For example, for quadratic spline blending functions $(m = 2)$ with four control points, i.e., $n + 1 = 4$ or $n = 3$. (order of B-spline function), parameter range $2 \le t \le 4$ is considered.

## B-spline Blending Function Formulation

The general expression for the calculation of coordinate positions $P(t)$ on a B-spline curve may be expressed as

$$P(t) = \sum_{i=1}^{n+1} P_i . N_{i,m}(t) \qquad t_{\min} \le t \le t_{\max} \qquad 2 \le m \le n+1$$

$P_i$ = Position vectors (coordinates) of $(n + 1)$ vertices defining the polygon control points

$N_{i,m}(t)$ = **Normalized B-spline blending (basis) functions**

# B-SPLINE BLENDING FUNCTION FORMULATION

$m =$ Order of B-spline blending functions

$m-1 =$ Degree of polynomials of B-spline blending functions

B-spline curve is defined as a polynomial spline function of order $m$ (hence, degree $m-1$ ) because it satisfies the following two conditions:

- B-spline function $P(t)$ is a polynomial of degree $(m-1)$ on each interval $x_i \le t \le x_{i+1}$
- B-spline function $P(t)$ and its derivatives of order 1, 2, 3 …$(m-2)$ are all continuous over the entire curve

# B-SPLINE BLENDING FUNCTION FORMULATION

Thus,

I. Fourth order $(m=4)$ B-spline curve is a piecewise cubic $(m-1)$ spline blending function (e.g., $at^3+bt^2+ct+d$) and curve possesses second order $(m-2)$, i.e. $C^2$ continuity.

II. Third order $(m=3)$ B-spline curve is a piecewise quadratic $(m-1)$ spline blending function (e.g., $at^2+bt+c$) and curve possesses first order $(m-2)$, i.e. $C^1$ continuity.

III. Second order $(m=2)$ B-spline curve is a piecewise linear $(m-1)$ spline blending function (e.g., $at+b$) and curve possesses zero order $(m-2)$, i.e. $C^0$ continuity.

IV. First order $(m=1)$ B-spline curve is a piecewise zero degree $(m-1)$ spline blending function; hence, the curve is just a point plot of the control point.

# B-SPLINE BLENDING FUNCTION FORMULATION

The blending function for B-spline curve is defined by the recursive formula

$$N_{i,m}(t) = \frac{(t - x_i)}{x_{i+m-1} - x_i} N_{i,m-1}(t) + \frac{(x_{i+m} - t)}{x_{i+m} - x_{i+1}} N_{i+1,m-1}(t)$$

B-spline blending function of order one $(m = 1)$ by Cox-deBoor as

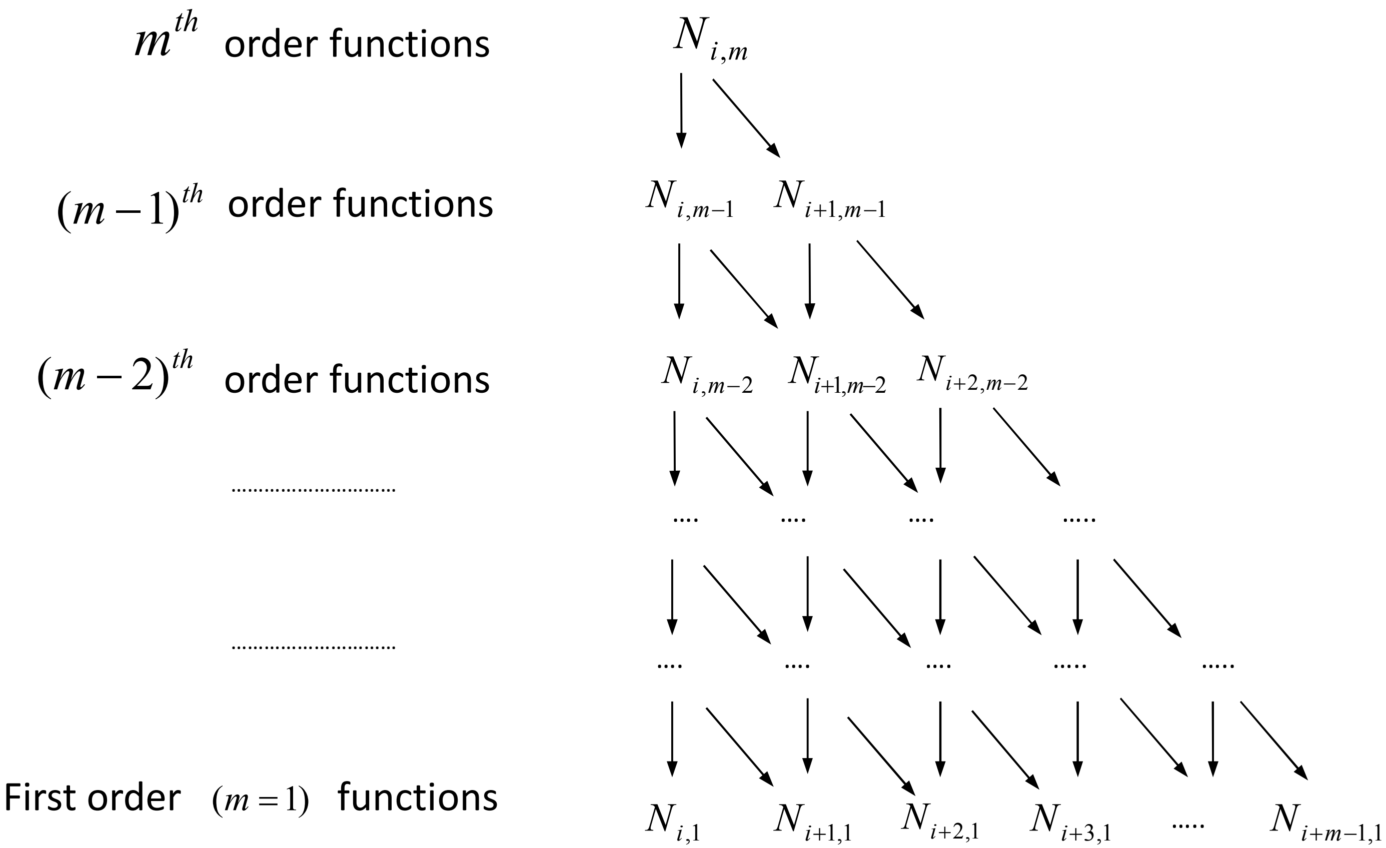
$$N_{i,1}(t) = \begin{cases} 1 & \text{if } x_i \le t \le x_{i+1} \\ 0 & \text{otherwise} \end{cases}$$

The Cox-deBoor formula is used to calculate B-spline blending functions in a recursive relation, a blending function of a *given order m depends on the lower order blending functions down to order 1*

For given blending function $N_{i,m}$ , Figure shows that this dependency on lower order blending functions forms a **triangular pattern**. Each higher order B-spline blending function, finally depends on the blending functions of order one.

# B-SPLINE BLENDING FUNCTION FORMULATION

## Dependency Diagram for Blending Function....



# COMPUTER AIDED DESIGN
# (BME-42)

## Unit-III: Space Curves

**(7 Lectures)**

- Properties for curve design, Parametric continuity,
- Parametric representation of synthetic curves, Spline curves and specifications, Parametric representation of synthetic curves
- Hermite curves-Blending functions formulation, shape control, properties,
- Bezier curves-Blending functions formulation, properties, Composite Bezier curves,
- Non-rational B-spline curves- Blending functions formulation, **knot vector, B-spline blending functions, properties**

# Lecture 25

## Topics Covered

**Non-Rational B-Spline Curves**

**Knot Vector**

**Types of Knot Vector**

Periodic Uniform Knot Vector

Open Uniform Knot Vector

Non-uniform Knot Vector

*Prepared By*

**Prof. S. K. SRIVASTAVA**

MED, MMMUT, Gorakhpur (UP)
sksme@mmmut.ac.in

## Knot Vector

- Local control of B-spline curve is obtained by defining the spline blending functions over the $m$ subintervals of total range of parameter.
- The selected set of subinterval endpoints $x_i$ for the range of parameter $t$ is referred to as knot vector.
- The knot vector is selected such that

$$x_i \leq x_{i+1}$$

- Knot vectors are real numbers and monotonically increases in the range of parameter $t$

The general knot vector is defined as

$$[X] = \begin{bmatrix} x_1 & x_2 & x_3 & . & . & . & . & x_i & x_{i+1} & . & . \end{bmatrix}$$

Where $x_{\min}$ and $x_{\max}$ depends on the number of control points.

## Knot Vector…

- The choice of knot vector has a significant influence on the spline blending functions $N_{i,m}(t)$

The general knot vector is defined as

$$[X] = [x_1 \quad x_2 \quad x_3 \quad x_4 \quad x_5 \quad . \quad . \quad . \quad . \quad . \quad .] = [0 \quad 1 \quad 2 \quad 3 \quad 4 \quad . \quad . \quad . \quad . \quad . \quad .]$$

In the following, spline blending functions of various *orders* have been determined:

### Spline blending functions of order 1

$m = 1$ means *one* subinterval for each blending function.

Degree of spline blending function, $m - 1 = 0$

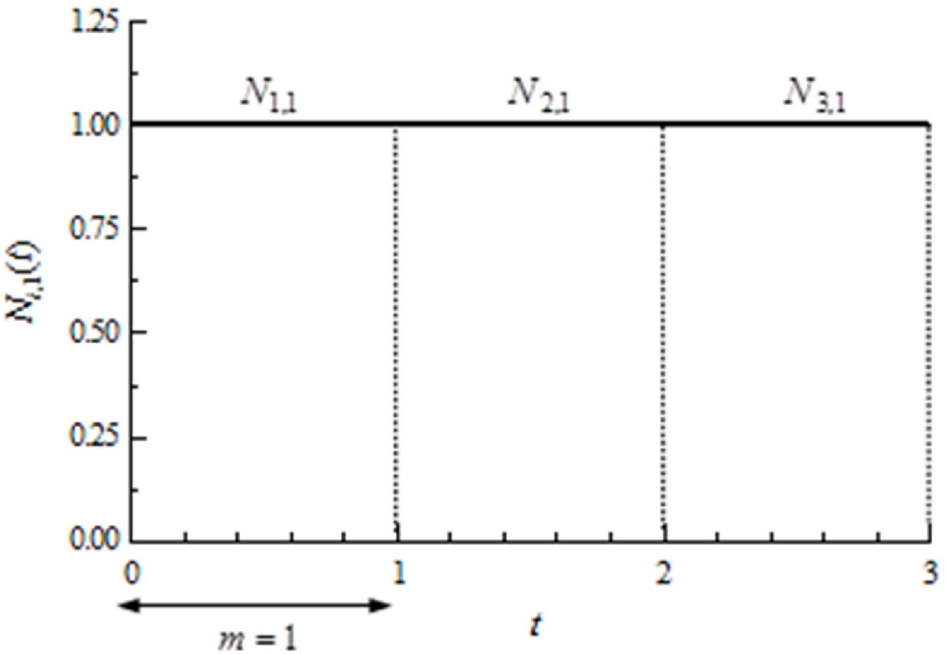
Hence, from blending function eqn., we have $N_{i,1}(t) = 1$ for $x_i \leq t \leq x_{i+1}$

## Knot Vector…

Therefore, successive blending functions of order 1 are given by

$$N_{1,1}(t) = N_{2,1}(t) = N_{3,1}(t) = \ldots\ldots = 1 \text{ (constant) for } x_i \le t \le x_{i+1}$$

Figure shows three *constant* blending functions.




**Spline blending functions for *m* = 1 and** $[X] = [0 \quad 1 \quad 2 \quad 3 \quad . \quad . \quad .]$

## Knot Vector…

### Spline blending functions of order 2

$m = 2$ means *two* subintervals for each blending function.

Degree of spline blending function, $m - 1 = 1$ (linear)

Hence, from eqn. we have

$$N_{i,m}(t) = \frac{(t - x_i)}{x_{i+m-1} - x_i} N_{i,m-1}(t) + \frac{(x_{i+m} - t)}{x_{i+m} - x_{i+1}} N_{i+1,m-1}(t)$$

For $i = 1$, $m = 2$, we have

$$N_{1,2}(t) = \frac{(t - x_1)}{x_2 - x_1} N_{1,1}(t) + \frac{(x_3 - t)}{x_3 - x_2} N_{2,1}(t)$$

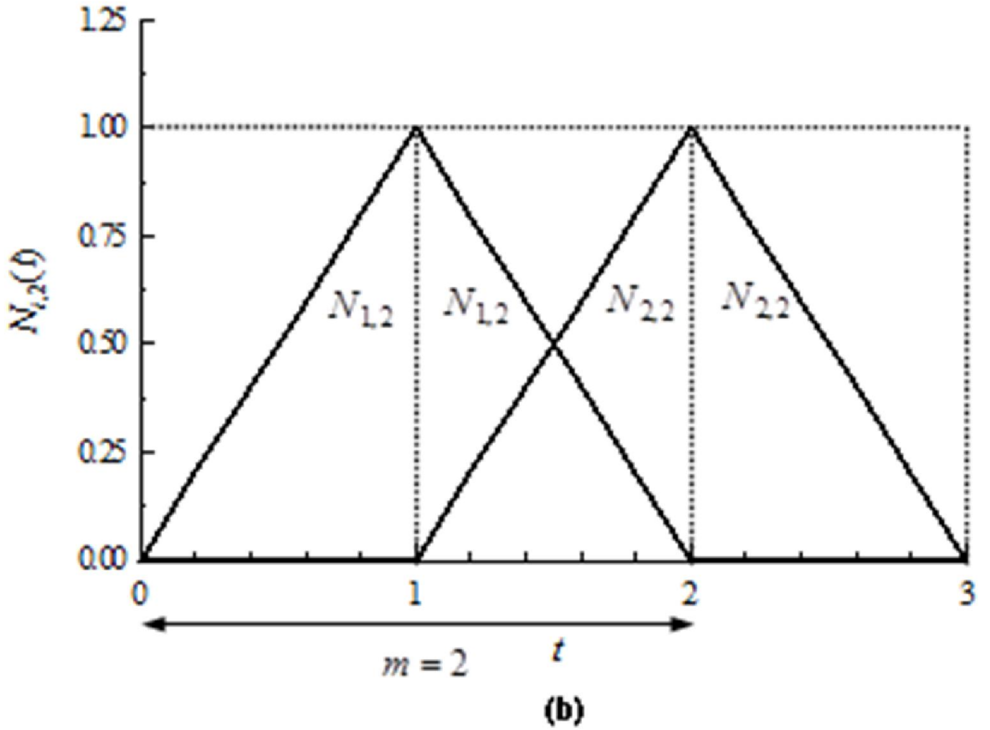
$$= \frac{(t - 0)}{1 - 0} N_{1,1}(t) + \frac{(2 - t)}{2 - 1} N_{2,1}(t)$$

or $N_{1,2}(t) = t.N_{1,1}(t) + (2 - t).N_{2,1}(t)$ (triangular shape)

Similarly, $N_{2,2}(t) = (t - 1).N_{2,1}(t) + (3 - t).N_{3,1}(t)$ (triangular shape shifted at $t = 1$)

## Knot Vector…

Figure shows the shape of two ***triangular*** shape blending functions.




**Spline blending functions for *m* = 2 and** $[X] = [0 \quad 1 \quad 2 \quad 3 \quad . \quad . \quad .]$

## Knot Vector…

### Spline blending functions of order 3

$m = 3$ means *three* subintervals for each blending function.

Degree of spline blending function, $m - 1 = 2$ (quadratic)

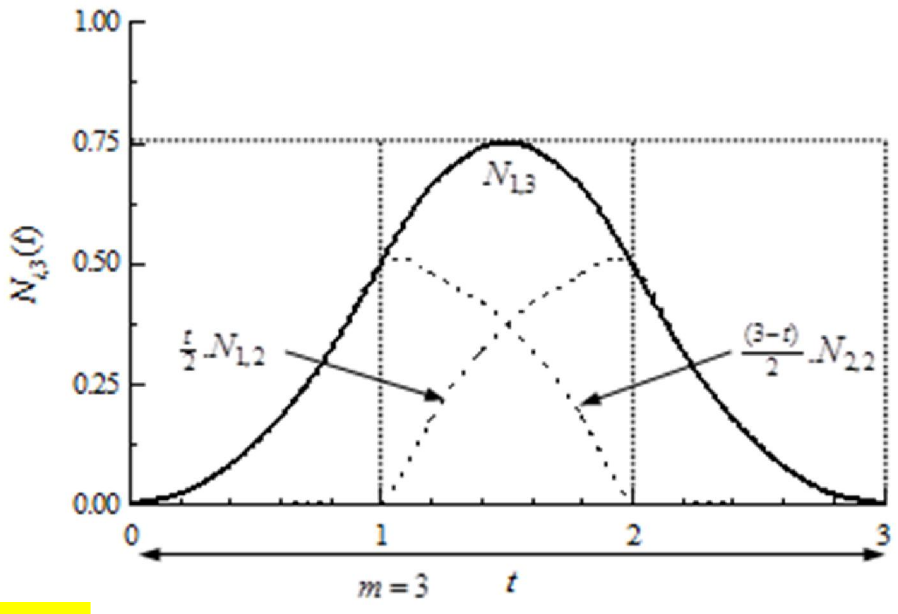
For $i = 1$, $m = 3$, eqn. gives

$$N_{1,3}(t) = \frac{(t - x_1)}{x_3 - x_1} N_{1,2}(t) + \frac{(x_4 - t)}{x_4 - x_2} N_{2,2}(t) = \frac{t}{2} N_{1,2}(t) + \frac{3 - t}{2} N_{2,2}(t)$$

Figure shows the shape of *quadratic* blending function.

**Spline blending functions for**

***m* = 3 and** $[X] = [0 \quad 1 \quad 2 \quad 3 \quad . \quad . \quad .]$




(c)

# TYPES OF KNOT VECTOR

## 1. Periodic Uniform Knot Vector

$$[X] = [0 \quad 1 \quad 2 \quad 3 \quad 4 \quad 5 \quad 6 \quad . \quad . \quad . \quad .]$$

$$= [-0.3 \quad -0.2 \quad -0.1 \quad 0 \quad 0.1 \quad 0.2 \quad 0.3 \quad . \quad . \quad . \quad .]$$

In general, uniform knot vector starts at 0 to some maximum value with an equal increment of 1, e.g.

$$[X] = [0 \quad 1 \quad 2 \quad 3 \quad 4]$$

In normalized form, we have

$$[X] = [\tfrac{0}{4} \quad \tfrac{1}{4} \quad \tfrac{2}{4} \quad \tfrac{3}{4} \quad \tfrac{4}{4}]$$

$$= [0 \quad 0.25 \quad 0.5 \quad 0.75 \quad 1]$$

For given order of spline blending function $(m)$, uniform knot vector results into *periodic uniform blending functions*, i.e., each blending function is a translate of the other.

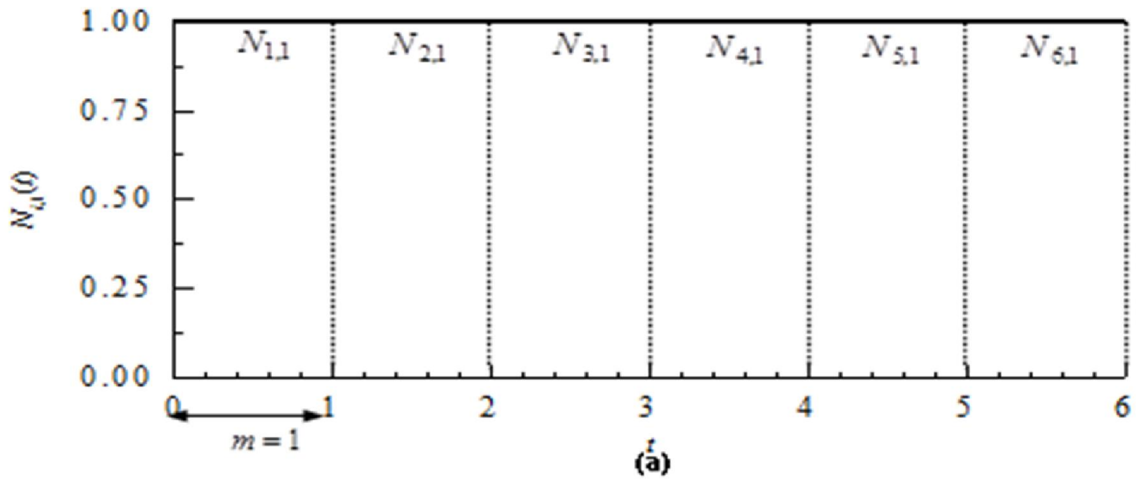
## 1. Periodic Uniform Knot Vector…

Mathematically, it is expressed as

$$N_{i,m}(t) = N_{i-1,m}(t-1) = N_{i+1,m}(t+1)$$

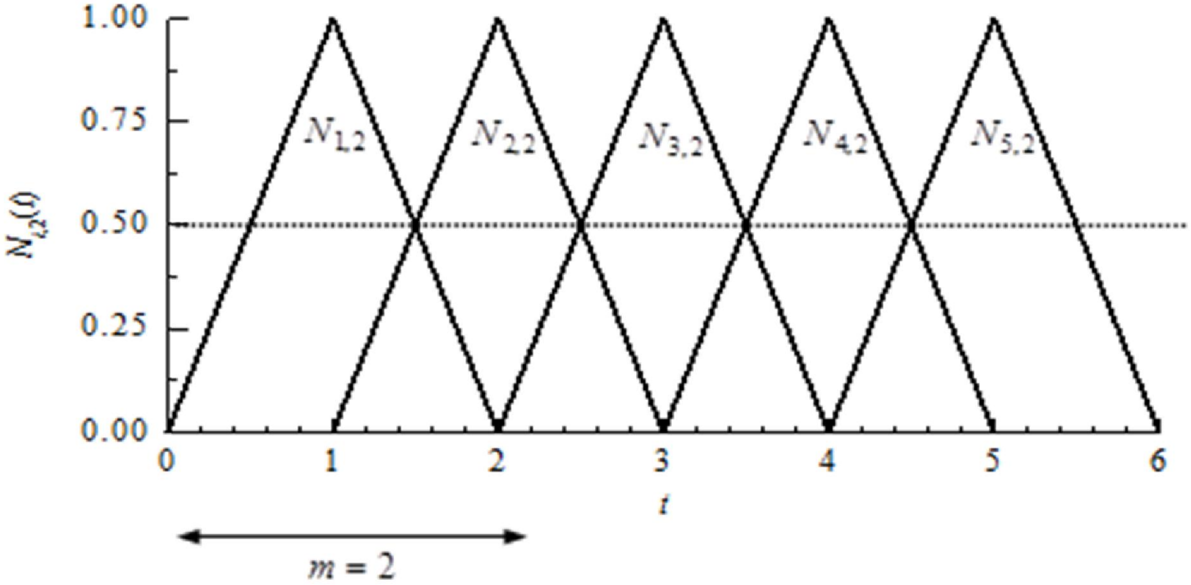
For example, for $m = 3$ (number of subintervals) and four control points $(n+1 = 4 \text{ or } n = 3)$, the number of knot values are $n + m + 1 = 3 + 3 + 1 = 7$; hence, seven knot vectors are expressed as

$$[X] = [x_1 \quad x_2 \quad x_3 \quad x_4 \quad x_5 \quad x_6 \quad x_7] = [0 \quad 1 \quad 2 \quad 3 \quad 4 \quad 5 \quad 6]$$




**Periodic uniform spline blending functions for *m* = 1, n + 1 = 4,** $[X] = [0 \quad 1 \quad 2 \quad 3 \quad 4 \quad 5 \quad 6]$,

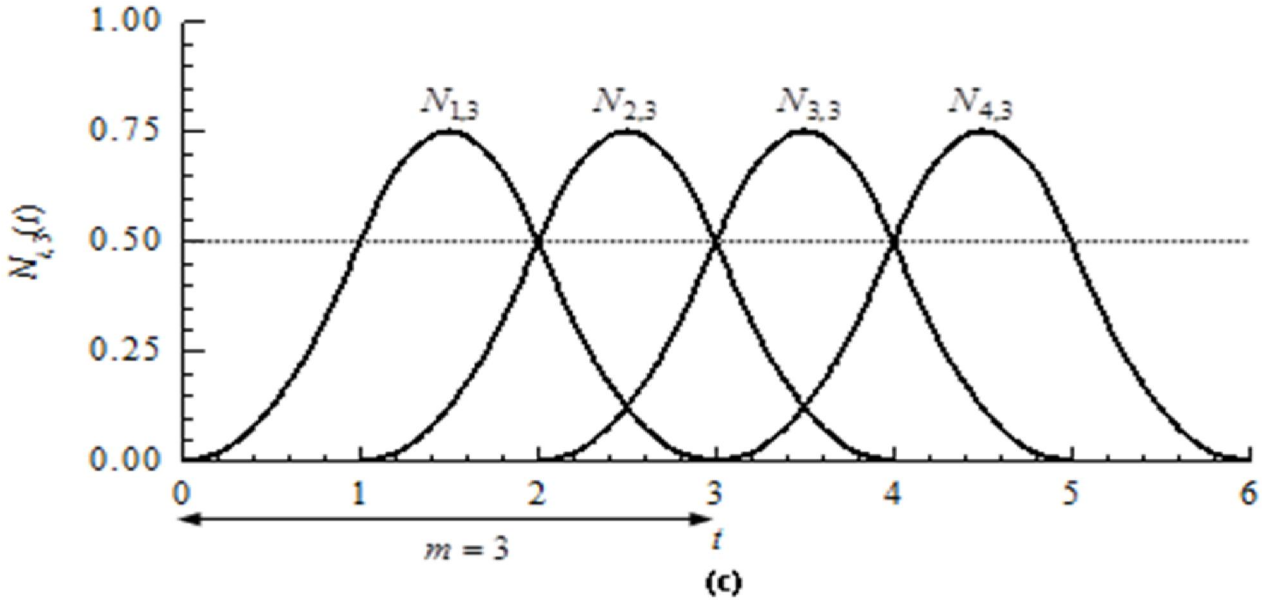
### 1. Periodic Uniform Knot Vector…




**Periodic uniform spline blending functions for *m* = 2, n + 1 = 4,**

$$[X] = [0 \quad 1 \quad 2 \quad 3 \quad 4 \quad 5 \quad 6],$$

## 1. Periodic Uniform Knot Vector…




**Periodic uniform spline blending functions for *m* = 3, n + 1 = 4,**

$$[X] = [0 \quad 1 \quad 2 \quad 3 \quad 4 \quad 5 \quad 6],$$

## 2. Open Uniform Knot Vector

- An open uniform knot vector has multiplicity of knots.
- The ends knot values in open uniform knot vector is equal to the order *m* of the spline blending (basis) function and internal knot values are equidistant for the entire range of parameter *t*.
- In general, open uniform knot vector is defined as

$$x_i = 0 \qquad \text{for} \qquad 1 \le i \le m$$

$$x_i = i - m \qquad \text{for} \qquad m+1 \le i \le n+1$$

$$x_i = n - m + 2 \qquad \text{for} \qquad n+2 \le i \le n+m+1$$

Table shows the calculations of general open uniform knot vector corresponding to the different order of spline blending functions.

## 2. Open Uniform Knot Vector…

**Table** Determination of open uniform knot vector for different order of spline blending functions

| Number of vertices $(n+1)$ | Order of spline functions $(m)$ | $x_i = 0$ <br> $1 \le i \le m$ | $x_i = i - m$ <br> $m+1 \le i \le n+1$ | $x_i = n - m + 2$ <br> $n+2 \le i \le n+m+1$ |
|---|---|---|---|---|
| 5 | 2 | $x_1 = 0$ <br> $x_2 = 0$ | $x_3 = i - m = 3 - 2 = 1$ <br> $x_4 = i - m = 4 - 2 = 2$ <br> $x_5 = i - m = 5 - 2 = 3$ | $x_6 = n - m + 2 = 4 - 2 + 2 = 4$ <br> $x_7 = n - m + 2 = 4 - 2 + 2 = 4$ |
| 5 | 3 | $x_1 = 0$ <br> $x_2 = 0$ <br> $x_3 = 0$ | $x_4 = i - m = 4 - 3 = 1$ <br> $x_5 = i - m = 5 - 3 = 2$ | $x_6 = n - m + 2 = 4 - 3 + 2 = 3$ <br> $x_7 = n - m + 2 = 4 - 3 + 2 = 3$ <br> $x_8 = n - m + 2 = 4 - 3 + 2 = 3$ |
| 5 | 4 | $x_1 = 0$ <br> $x_2 = 0$ <br> $x_3 = 0$ <br> $x_4 = 0$ | $x_5 = i - m = 5 - 4 = 1$ | $x_6 = n - m + 2 = 4 - 4 + 2 = 2$ <br> $x_7 = n - m + 2 = 4 - 4 + 2 = 2$ <br> $x_8 = n - m + 2 = 4 - 4 + 2 = 2$ <br> $x_9 = n - m + 2 = 4 - 4 + 2 = 2$ |

## 2. Open Uniform Knot Vector…

For integer increments, the open uniform knot vectors are calculated as

$m = 2$ (multiplicity of two-knot values), $[X] = [0 \quad 0 \quad 1 \quad 2 \quad 3 \quad 4 \quad 4]$

$m = 3$ (multiplicity of three-knot values), $[X] = [0 \quad 0 \quad 0 \quad 1 \quad 2 \quad 3 \quad 3 \quad 3]$

$m = 4$ (multiplicity of four-knot values), $[X] = [0 \quad 0 \quad 0 \quad 0 \quad 1 \quad 2 \quad 2 \quad 2 \quad 2]$

In normalized form, we have

$m = 2, \quad [X] = [0 \quad 0 \quad \tfrac{1}{4} \quad \tfrac{1}{2} \quad \tfrac{3}{4} \quad 1 \quad 1]$

$m = 3, \quad [X] = [0 \quad 0 \quad 0 \quad \tfrac{1}{3} \quad \tfrac{2}{3} \quad 1 \quad 1 \quad 1]$

$m = 4, \quad [X] = [0 \quad 0 \quad 0 \quad 0 \quad \tfrac{1}{2} \quad 1 \quad 1 \quad 1 \quad 1]$

# TYPES OF KNOT VECTOR…

## 2. Open Uniform Knot Vector…

*When number of defining polygon vertices is equal to the order of spline blending functions, and an open uniform knot vector is used, the spline blending functions are reduced to Bernstein polynomials. Alternatively, B-spline curve converts into a Bézier curve.*

Mathematically, when $n+1=m$, the open uniform knot vector results a Bézier curve. For example, for $n+1=m=4$, the knot vector values are calculated as

$x_i = 0$ for $1 \le i \le m$, i.e., $x_1 = x_2 = x_3 = x_4 = 0$ for $1 \le i \le 4$

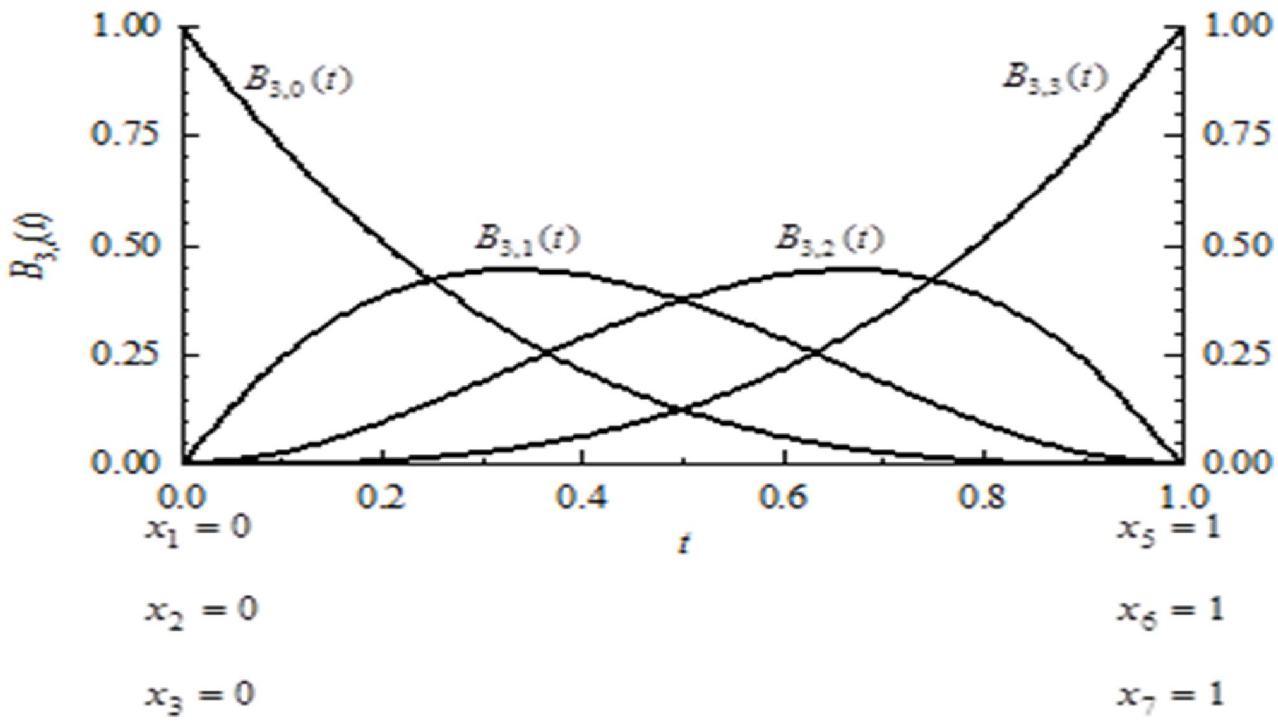
$x_i = n-m+2$ for $n+2 \le i \le n+m+1$, i.e., $x_5 = x_6 = x_7 = x_8 = n-m+2 = 3-4+2 = 1$

Therefore, knot vector is just $m$ *zeros* followed by $m$ *ones*, thus

$$[X] = [x_1 \quad x_2 \quad x_3 \quad x_4 \quad x_5 \quad x_6 \quad x_7 \quad x_8] = [0 \quad 0 \quad 0 \quad 0 \quad 1 \quad 1 \quad 1 \quad 1]$$

## 2. Open Uniform Knot Vector…

This results into a cubic Bézier/B-spline curve. Fig. shows the corresponding Bézier/B-spline blending functions for four control points, $n + 1 = m = 4$.




$x_4 = 0$ $x_8 = 1$

## 2. Open Uniform Knot Vector…

### B-Spline curve of third order

- Let, the order of spline blending function is 3 ( m = 3); therefore, degree of blending function is quadratic.
- For m = 3 (three subintervals for each blending function), if multiplicity of three at origin occurs the knot vector for five control points may be obtained as given below:

Five control points $P_1, P_2, P_3, P_4, P_5$

Multiplicity at endpoints : $m=3$ Control points : $n+1=5$ or $n=4$

Parameter range : $0 \le t \le n-m+2$ , i.e. $0 \le t \le 3$

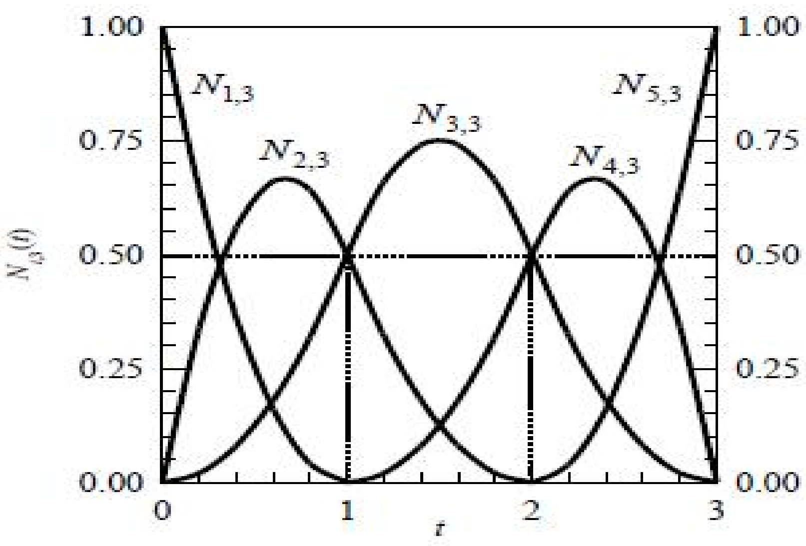
Number of knot values = $n+m+1$ , i.e. 4+3+1 = 8

Knot vector: $[X]=[x_1 \quad x_2 \quad x_3 \quad x_4 \quad x_5 \quad x_6 \quad x_7 \quad x_8]=[0 \quad 0 \quad 0 \quad 1 \quad 2 \quad 3 \quad 3 \quad 3]$

The B-spline curve $P(t)=\sum_{i=1}^{n+1} P_i.N_{i,m}(t)=\sum_{i=1}^{5} P_i.N_{i,3}(t)=P_1N_{1,3}+P_2N_{2,3}+P_3N_{3,3}+P_4N_{4,3}+P_5N_{5,3}$

## 2. Open Uniform Knot Vector…

B-Spline curve of third order



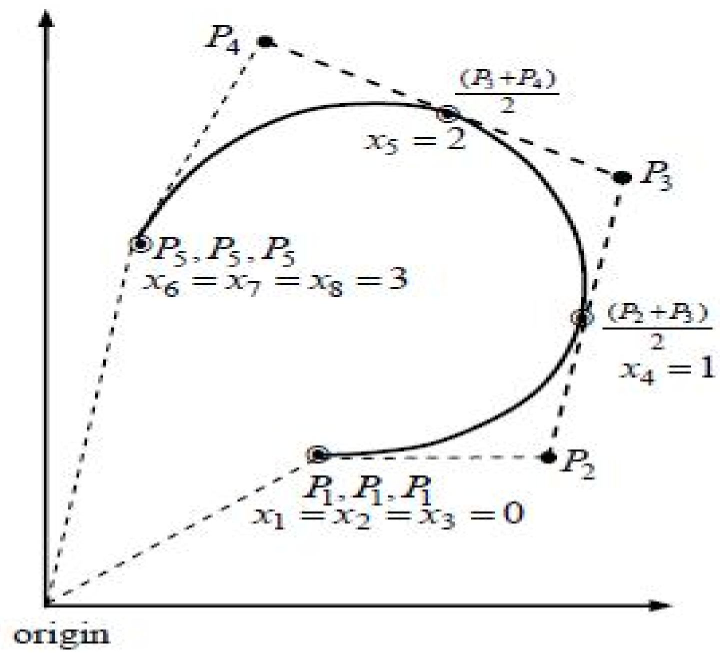




Fig. shows the B-spline curve for five control points. The curve interpolates the end control points $P_1$ and $P_5$ because it has multiplicity of 3 (equal to the order of spline blending function) at the end points.

### 3. Non-uniform Knot Vector

- For this class of splines, any value and spacing between the knots can be specified.
- When internal knots are unequally spaced or have multiple values, it results into nonuniform blending functions.
- The unequally spaced knot values results into different shapes of B-spline blending functions for different intervals.
- The increase in multiplicity of knot values introduces substantial modifications in the shape of curve and even introduces discontinuities.

Some examples of nonuniform knot vectors are

$$[X] = [0 \quad 0.27 \quad 0.5 \quad 0.74 \quad 1]$$

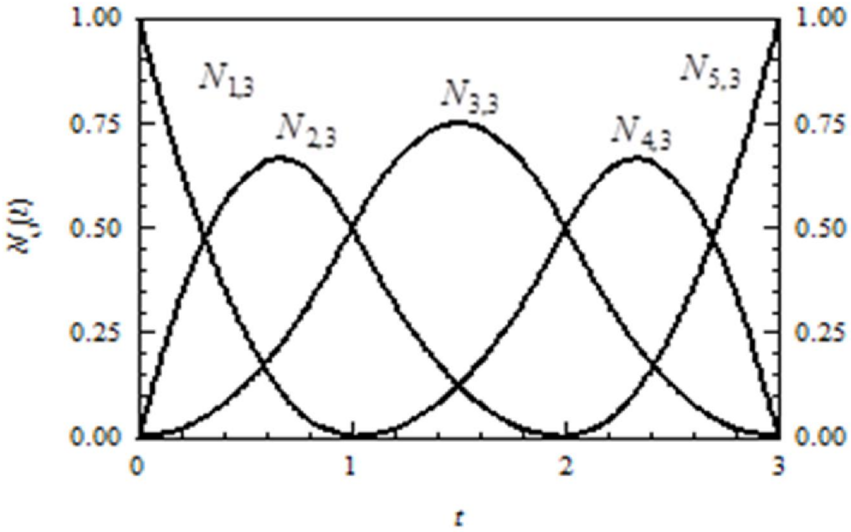
$$[X] = [0 \quad 0 \quad 0 \quad 1.7 \quad 2.3 \quad 3 \quad 3 \quad 3]$$

$$[X] = [0 \quad 0 \quad 0 \quad 1 \quad 1 \quad 3 \quad 3 \quad 3]$$

$$[X] = [0 \quad 0 \quad 0 \quad 1.7 \quad 2.3 \quad 3 \quad 3 \quad 3]$$

# TYPES OF KNOT VECTOR…

### 3. Non-uniform Knot Vector…

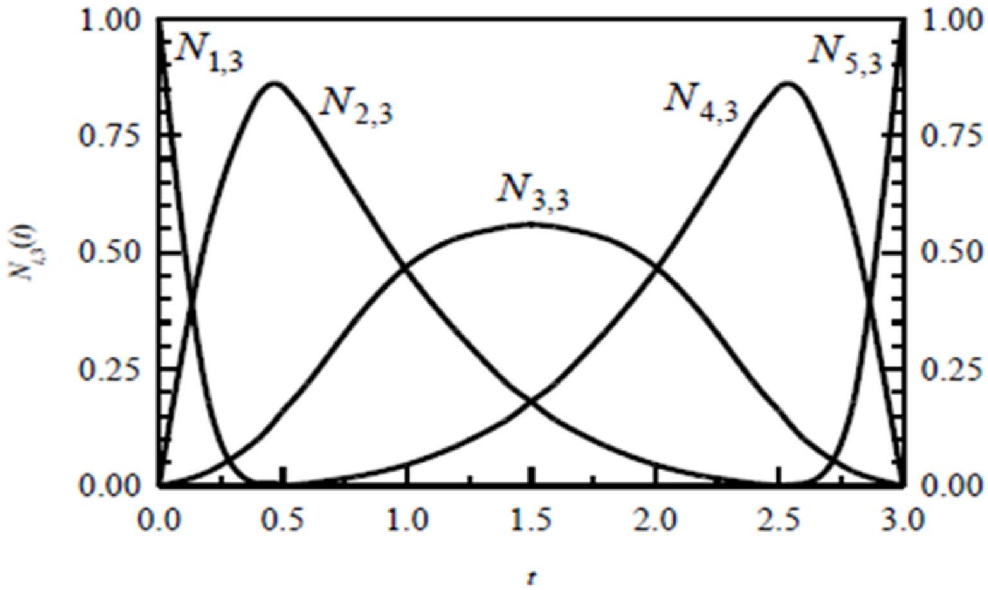



Nonuniform blending functions with unequally spaced internal knots for $n+1=5$ , $m=3$ .

$$[X]=[0 \quad 0 \quad 0 \quad 1 \quad 2 \quad 3 \quad 3 \quad 3]$$

***The nonuniform knot vectors have multiplicity of three at the ends***

### 3. Non-uniform Knot Vector…




Nonuniform blending functions with unequally spaced internal knots for $n+1=5$, $m=3$.

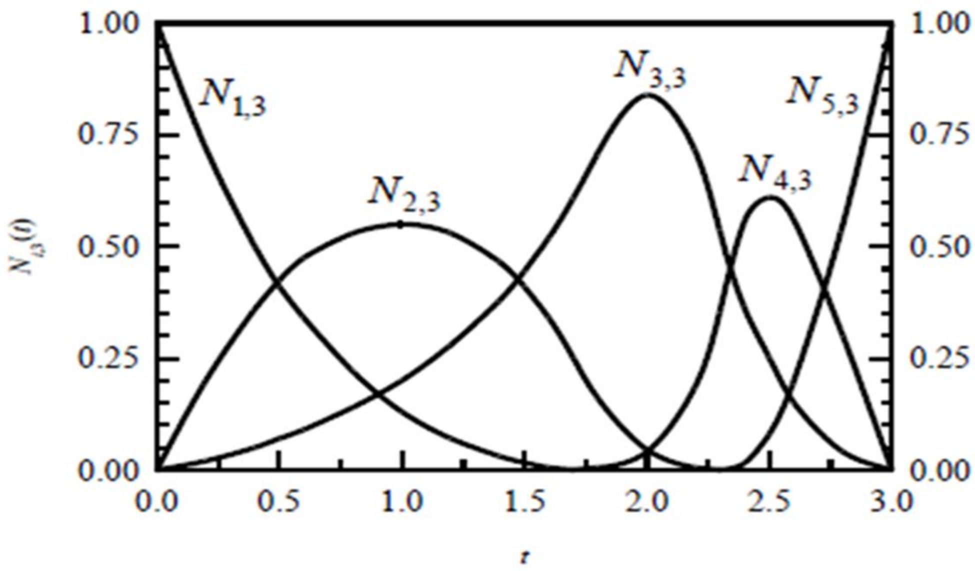
$$[X]=[0 \quad 0 \quad 0 \quad 0.5 \quad 2.5 \quad 3 \quad 3 \quad 3]$$

*Any value and spacing between the knots.*

***The nonuniform knot vectors have multiplicity of three at the ends***

# TYPES OF KNOT VECTOR…

## 3. Non-uniform Knot Vector…



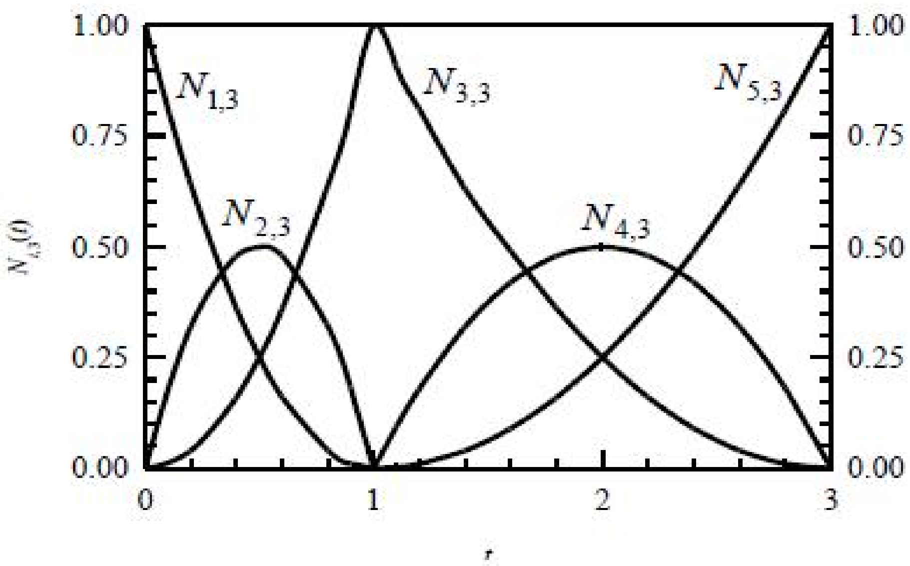

Nonuniform blending functions with unequally spaced internal knots for $n+1=5$ , $m=3$ .

$$[X]=[0 \quad 0 \quad 0 \quad 1.7 \quad 2.3 \quad 3 \quad 3 \quad 3]$$

*Any value and spacing between the knots.*

***The nonuniform knot vectors have multiplicity of three at the ends***

### 3. Non-uniform Knot Vector…



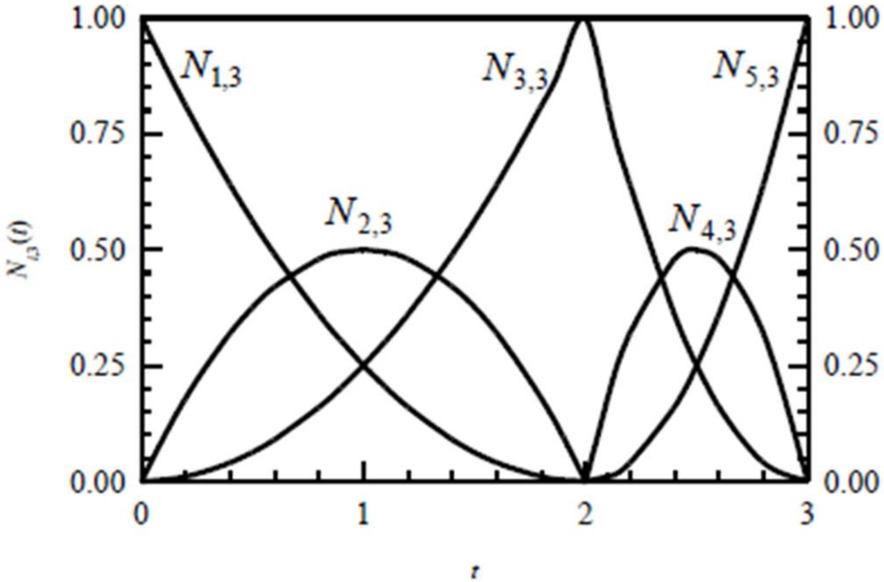

Nonuniform blending functions with multiple internal knots for $n+1=5$ , $m=3$ .

$$[X]=[0 \quad 0 \quad 0 \quad 1 \quad 1 \quad 3 \quad 3 \quad 3]$$

*Any value and spacing between the knots. Internal multiplicity occurs at knot value 1.*

*The nonuniform knot vectors have multiplicity of three at the ends*

### 3. Non-uniform Knot Vector…




Nonuniform blending functions with multiple internal knots for $n+1=5$ , $m=3$ .

$$[X] = [0 \quad 0 \quad 0 \quad 2 \quad 2 \quad 3 \quad 3 \quad 3]$$

*Any value and spacing between the knots. Internal multiplicity occurs at knot value 2.*

*The nonuniform knot vectors have multiplicity of three at the ends*

# COMPUTER AIDED DESIGN
## (BME-42)

### Unit-III: Space Curves
**(7 Lectures)**

- Properties for curve design, Parametric continuity,
- Parametric representation of synthetic curves, Spline curves and specifications, Parametric representation of synthetic curves
- Hermite curves-Blending functions formulation, shape control, properties,
- Bezier curves-Blending functions formulation, properties, Composite Bezier curves,
- Non-rational B-spline curves- Blending functions formulation, knot vector, B-spline blending functions, **properties**

# Lecture 26

### Topics Covered

**Shape Control of Non-Rational B-Spline Curves**

**Properties of Non-Rational B-Spline Curves**

*Prepared By*

Prof. S. K. SRIVASTAVA

MED, MMMUT, Gorakhpur (UP)
sksme@mmmut.ac.in

# SHAPE CONTROL OF NON-RATIONAL B-SPLINE CURVES

The following methods may be used for controlling the *shape* of nonrational B-spline curves.

I. Changing the type of knot vector, i.e., periodic uniform, open uniform and nonuniform, alternatively, the shape of spline blending functions.

II. Changing the number ($n + 1$) and location (i.e., knot values) of the defining polygon vertices.

III. Changing the order ($m$) of B-spline blending function, i.e., quadratic ($m = 2$), cubic ($m = 3$) and so on (Figure)

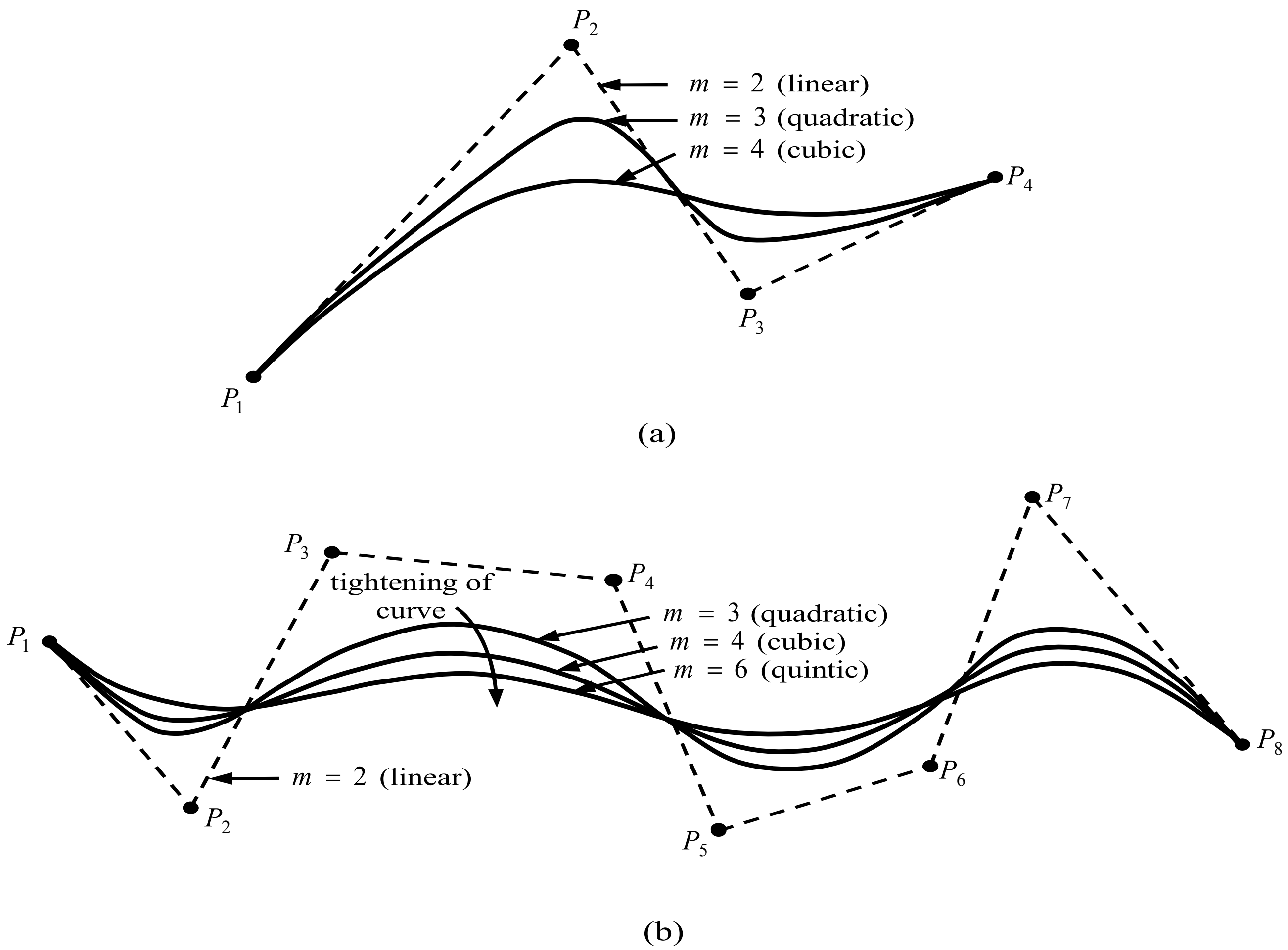
$m = 1$ results into zero degree curves, i.e., point plot of the control points.

$m = 2$ results into one-degree (linear) curve, i.e., polygon segments themselves.

$m = 3$ results into two-degree (quadratic) B-spline curve.

$m = 4$ results into three-degree (cubic) B-spline curve, i.e., a Bézier curve.

# SHAPE CONTROL OF NON-RATIONAL B-SPLINE CURVES




**Effect of varying order (degree) of B-spline curves on its Shape**

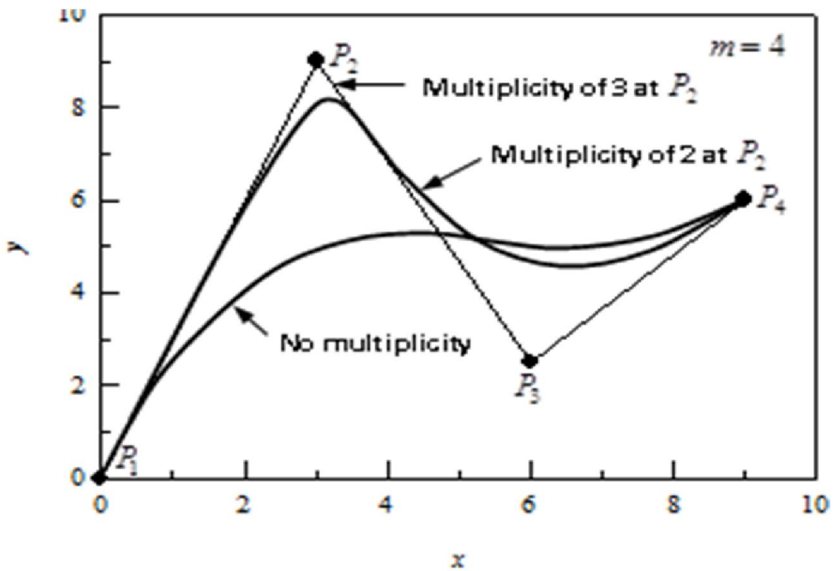
# SHAPE CONTROL OF NON-RATIONAL B-SPLINE CURVES

IV. Using the internal multiplicity and ends multiplicity of knots

The *internal multiplicity* induces a cusp in one of the blending functions; moreover, the location of cusp changes with the change in the values of multiple internal knot vectors

IV. Using multiplicity of polygon vertices

*Internal multiplicity* of knot values induces the regions of high curvature in B-spline curve




**All the curves are of fourth order ($m$ = 4)**

# SHAPE CONTROL OF NON-RATIONAL B-SPLINE CURVES

Lowest curve defined by the four polygon vertices $P_1, P_2, P_3, P_4$ with knot vector

$$[X] = [0 \quad 0 \quad 0 \quad 0 \quad 1 \quad 1 \quad 1 \quad 1]$$

Middle curve defined by the five polygon vertices $P_1, P_2, P_2, P_3, P_4$ (two coincident/multiple vertices at $P_2$) with knot vector

$$[X] = [0 \quad 0 \quad 0 \quad 0 \quad 1 \quad 2 \quad 2 \quad 2 \quad 2]$$

Highest curve (polygon itself) defined by the six polygon vertices $P_1, P_2, P_2, P_2, P_3, P_4$ (three coincident/multiple vertices at $P_2$) with knot vector

$$[X] = [0 \quad 0 \quad 0 \quad 0 \quad 1 \quad 2 \quad 3 \quad 3 \quad 3 \quad 3]$$

# SHAPE CONTROL OF NON-RATIONAL B-SPLINE CURVES

- The B-spline curve pulls more towards the control point $P_2$ by increasing its multiplicity
- A *sharp corner* can be created by keeping the number of multiple vertices equal to $m$-1.
- Thus, in the present situation, a sharp corner is created at the vertex using a multiplicity of 3 ($m$ - 1 = 4 - 1 = 3). Polygon itself has multiplicity of 3, i.e., $P_2, P_2, P_2$
- It should be noted that a linear segment occurs on both sides of multiple vertex

# PROPERTIES OF NON-RATIONAL B-SPLINE CURVES…

- The shapes of B-spline curves depend upon the shapes of spline blending functions.
- Alternatively, the properties of B-spline curves depend upon the properties of spline blending functions, and the way their shapes are controlled.
- It is very difficult to control and calculate the B-spline curve accurately when higher degree polynomials are used; therefore, cubic B-spline curves are generally preferred for large number of CAD applications.

The properties of B-spline functions are

1. The B-spline function $P(t)$ is a polynomial of degree $m-1$ (where $m$ is the order of spline blending functions) on each interval $x_i \le t \le x_{i+1}$.
2. The B-spline functions $P(t)$ and its derivatives of order 1, 2, 3, …., $m-2$ are all continuous over the entire range of parameter $t$.

# PROPERTIES OF NON-RATIONAL B-SPLINE CURVES…

3. Each spline blending functions has precisely one peak value except for $m = 1$, where the peak value is constant, i.e., '1' for the entire range of parameter $t$.

4. The sum of spline blending functions is unity for any value of parameter $t$ $(0 \le t \le 1)$. Mathematically, we have

$$\sum_{i=1}^{n+1} N_{i,m}(t) \equiv 1$$

Each blending function is either positive or zero for all values of parameter $t$, i.e. $N_{i,m}(t) \ge 0$.

5. Each blending function $N_{i.m}(t)$ is defined over $m$ subintervals of the total range of parameter $t$ for $x_i \le t \le x_{i+1}$.

6. If order of B-spline curve is equal to the number of defining polygon vertices (i.e., $m = n+1$), the spline blending functions are termed as Bernstein polynomials; consequently, B-spline curve reduces to a Bézier curve.

7. The curve generally follows the shape of the defining polygon.

# PROPERTIES OF NON-RATIONAL B-SPLINE CURVES…

8. The B-spline curve exhibits the variation diminishing property. Thus, the curve does not oscillate about any straight line more often than the sides of its characteristic polygon.

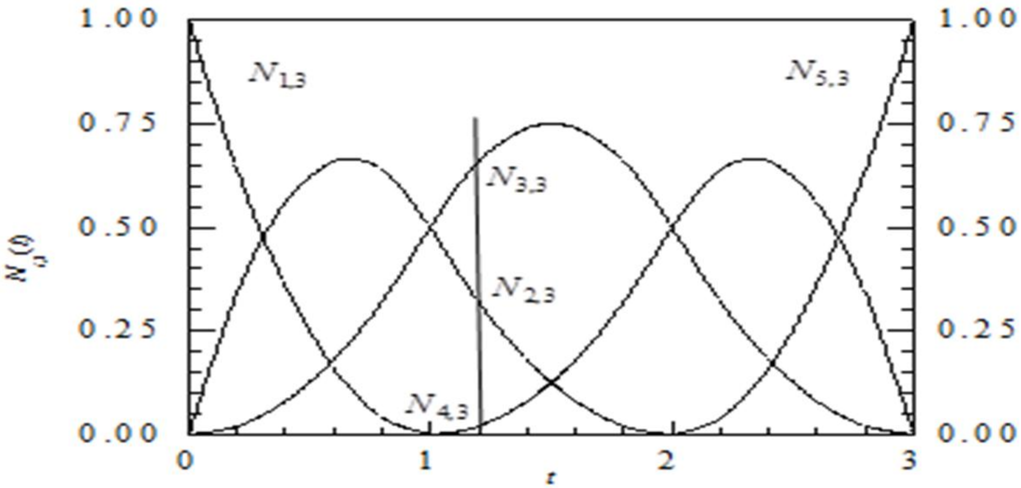
9. The B-spline curve is invariant under an affine transformation. An affine transformation is a combination of linear transformations, e.g., rotation followed by the translation. For an affine transformation, the last row in a general 4x4 transformation matrix is $[0 \quad 0 \quad 0 \quad 1]$. Any affine transformation can be applied to the curve by applying it to the defining polygon vertices, i.e., the curve is transformed by transforming the defining polygon vertices.

10. The B-spline curve is described with $n+1$ blending functions corresponding to the $n+1$ control points.

11. The range of parameter $t$ is divided into $m$ subintervals by the $n+m+1$ knot values specified in the knot vector.

12. Each section of B-spline curve is influenced by the $m$ control points. Conversely, one control point can affect the shape of the curve at most $m$ curve sections. For $t=1.2$, the curve is affected by $m=3$ control points corresponding to the spline blending functions $N_{2,3}$, $N_{3,3}$ and $N_{4,3}$.

# PROPERTIES OF NON-RATIONAL B-SPLINE CURVES…



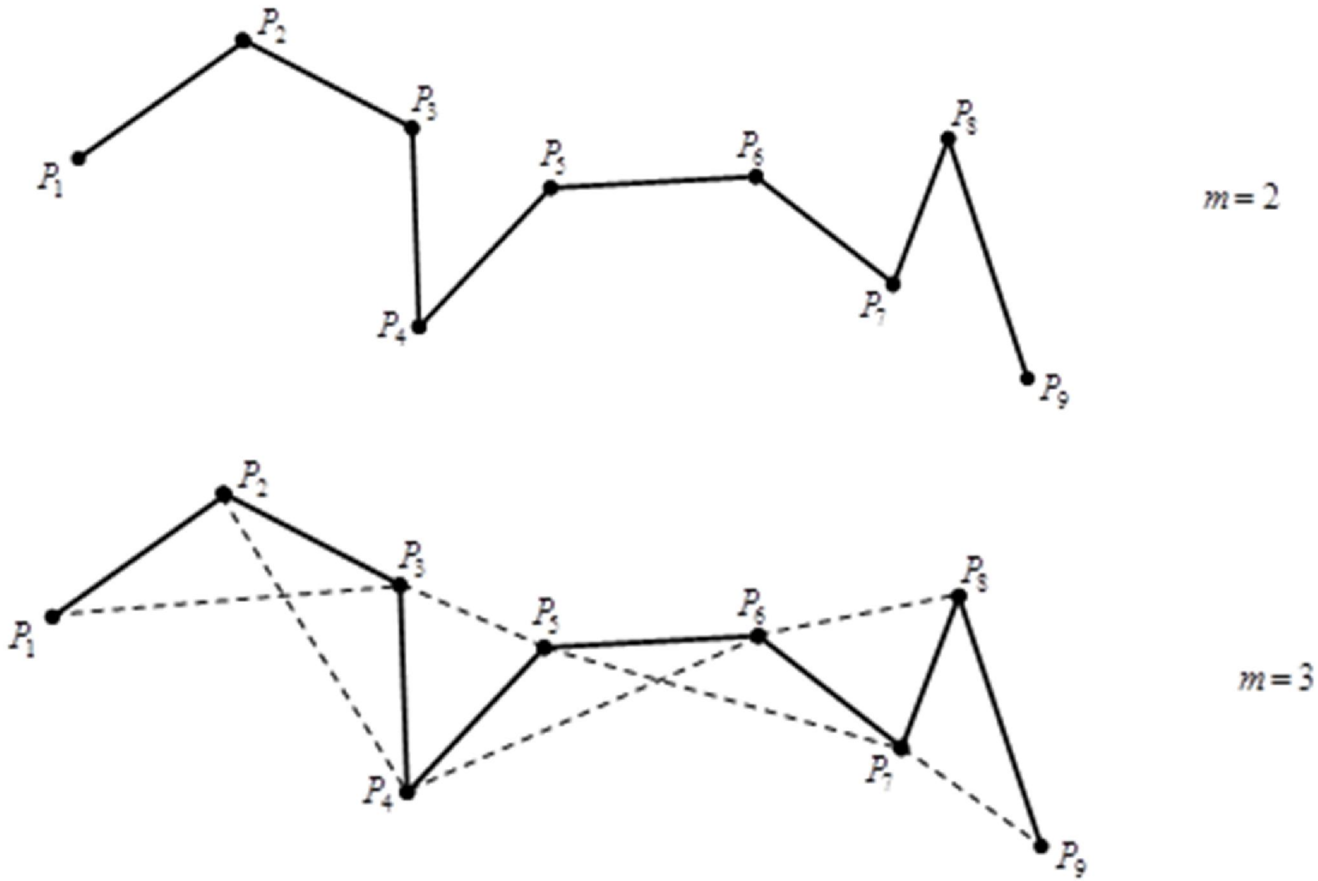

**Dependencies of higher order nonuniform spline blending functions on lower order blending functions for** $n+1=5$ **,** $m=3$ **,** $[X]=[0 \quad 0 \quad 0 \quad 2 \quad 2 \quad 3 \quad 3 \quad 3]$

# PROPERTIES OF NON-RATIONAL B-SPLINE CURVES…

13. The curve lies within the convex hull of its defining polygon.

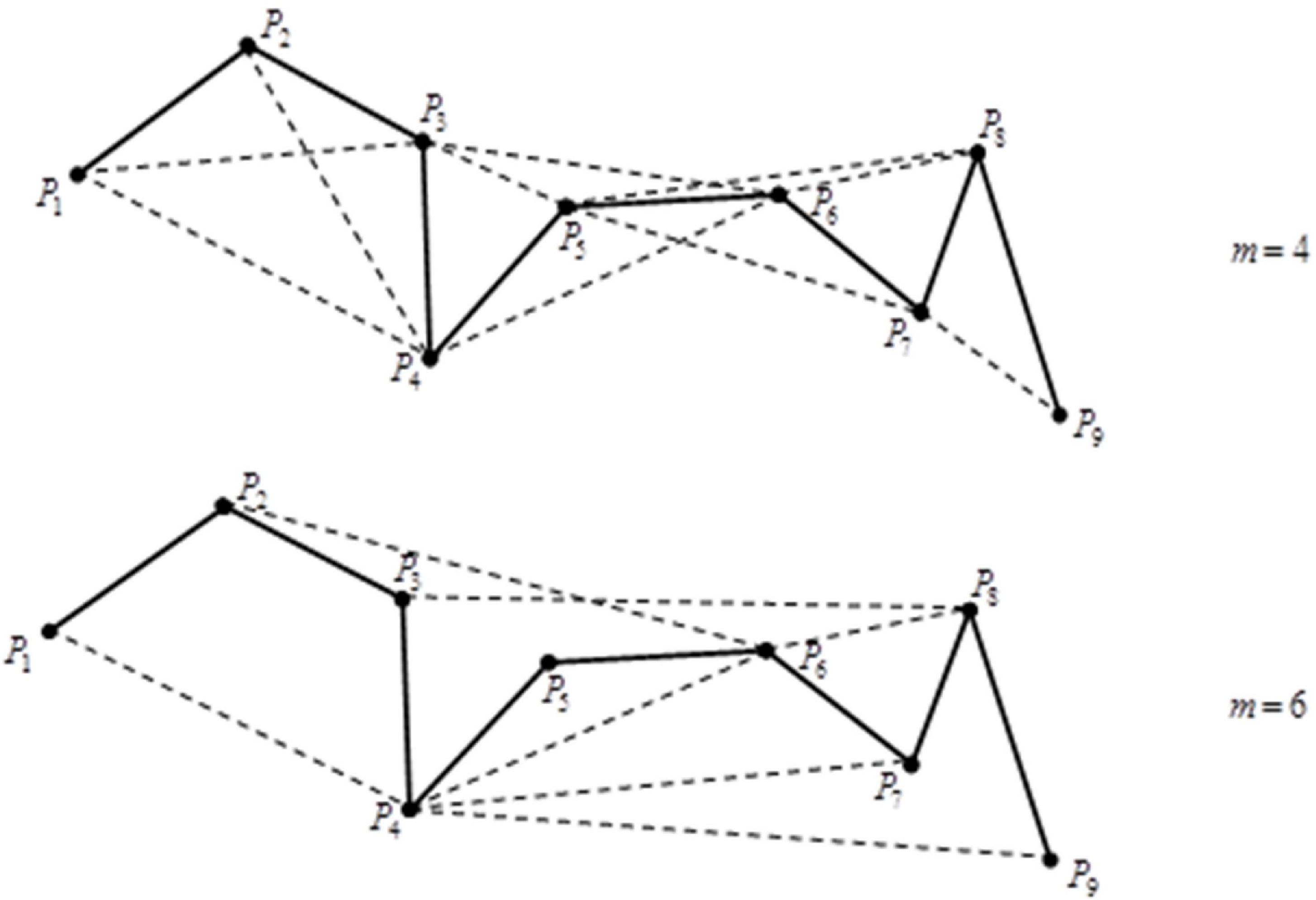
The convex hull property of B-spline curves is stronger than that of the Bézier curves. A point on B-spline curve must lie within the convex hull of $m$ successive control points. Therefore, all points on the curve must lie within the union of all convex hulls, formed by considering $m$ successive defining polygon vertices. Figure illustrates the convex hull of B-spline curves, shown within the dotted area of the polygons, of different order $m$. For $m = 2$, the convex hull is the defining polygon itself; however, the convex hull for higher order B-spline curves $(m = 3, 4, 6)$ is defined as the *union* of convex hulls of $m$ successive defining polygon vertices.

# PROPERTIES OF NON-RATIONAL B-SPLINE CURVES…




**Convex hull properties of B-spline curves (m = 2 and 3)**

# PROPERTIES OF NON-RATIONAL B-SPLINE CURVES…

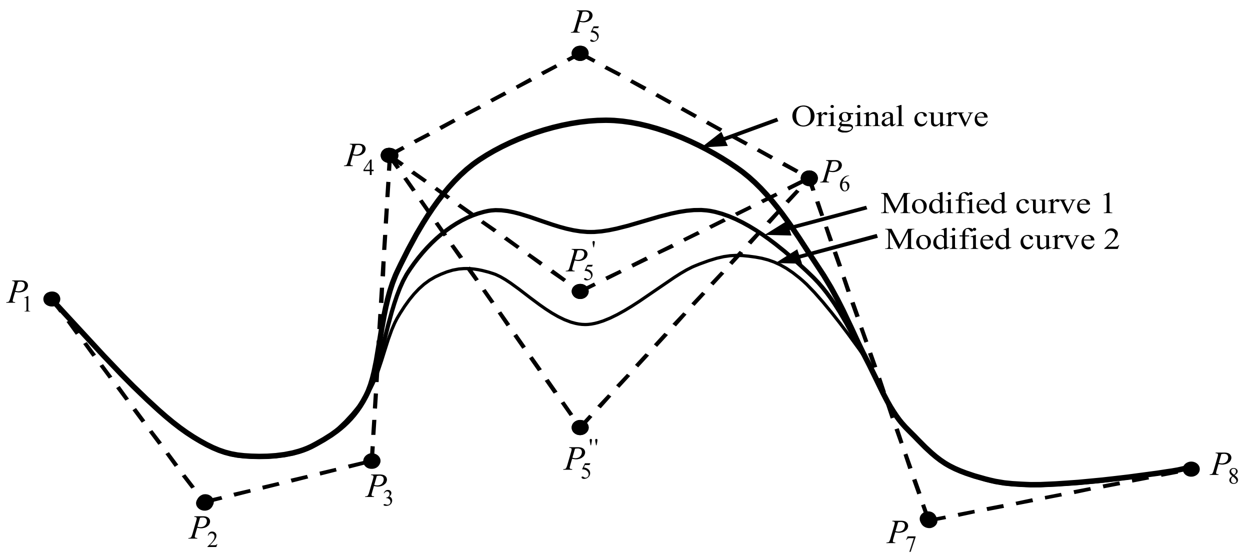



**Convex hull properties of B-spline curves (m = 4 and 6)**

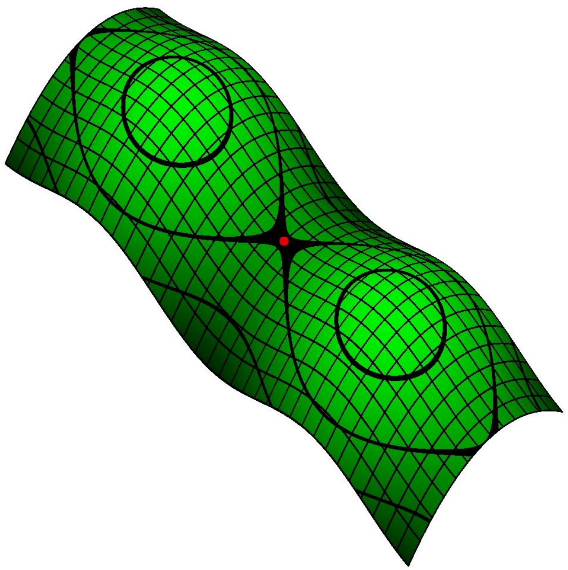
# PROPERTIES OF NON-RATIONAL B-SPLINE CURVES…

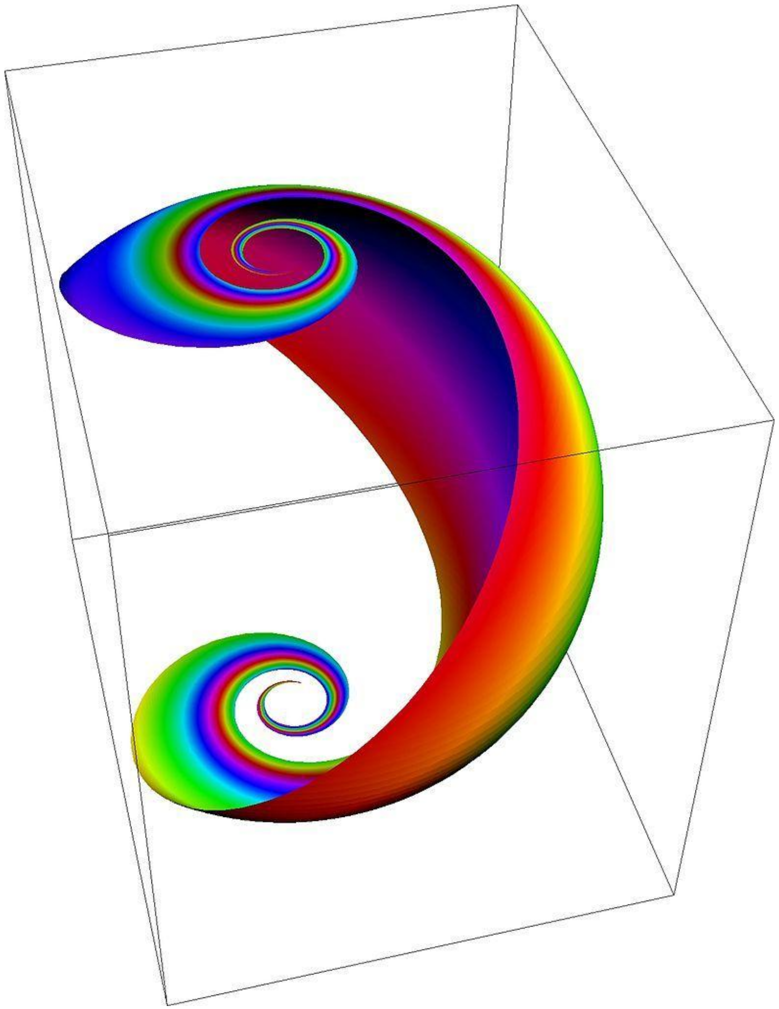
13. The B-spline curve tightens with an increase in the degree of the curve. Generally, lesser the degree, closer is the curve towards the control points. Fig. shows the effect of increase in order (degree) of B-spline functions on its shape.
14. Internal multiplicity of control points induces the regions of high curvature. Alternatively, the curve pulls more towards the polygon vertices by increasing its multiplicity. This property helps in generating the sharp corners in B-spline curves.
15. Local modifications over any B-spline curve segment are possible. In general, the curve is affected for the curve segments corresponding to $\pm m/2$ polygon spans around the displaced polygon vertex. Fig. 8.48 shows three B-spline curves, each of order four $(m = 4)$, obtained by moving vertex $P_5$, successively to the new locations $P_5^{'}$ and $P_5^{''}$ thereby depicting the local modifications (i.e., shape change in the limited region) in the original curve. When vertex $P_5$ shifts to $P_5^{'}$, only the curve segments corresponding to polygon spans $P_3P_4$, $P_4P_5$, $P_5P_6$ and $P_6P_7$ are affected, i.e., only two curve segments corresponding to the two polygon spans $(\pm m/2 = 4/2 = 2)$ around the displaced vertex $P_5$ are affected.

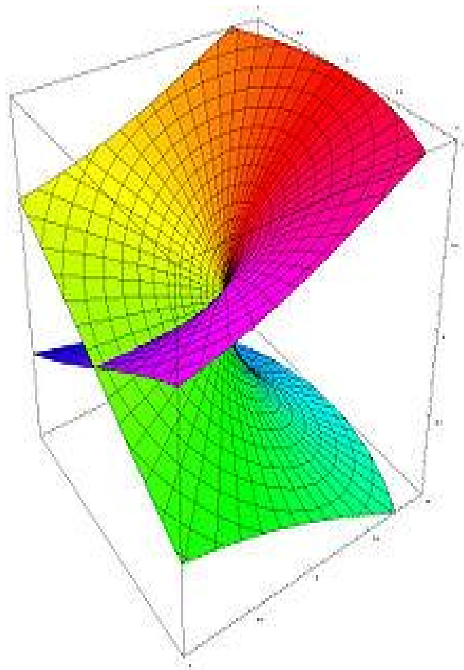
# PROPERTIES OF NON-RATIONAL B-SPLINE CURVES…




**Local control (modification) of B-spline curves**







# THANK YOU