16.810

Engineering Design and Rapid Prototyping

Lecture 3a

Computer Aided Design (CAD)

Instructor(s)

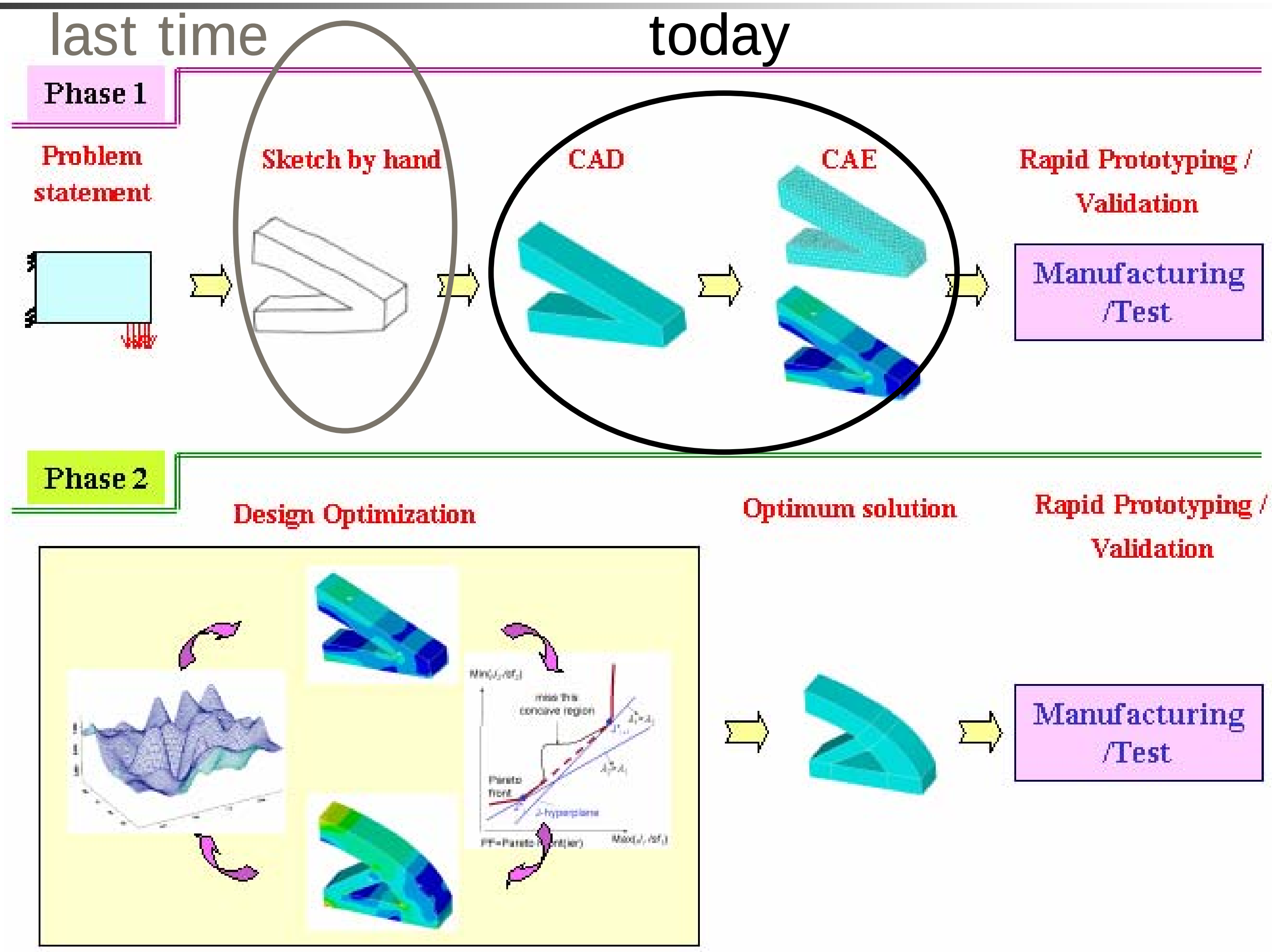
Prof. Olivier de Weck

January 16, 2007

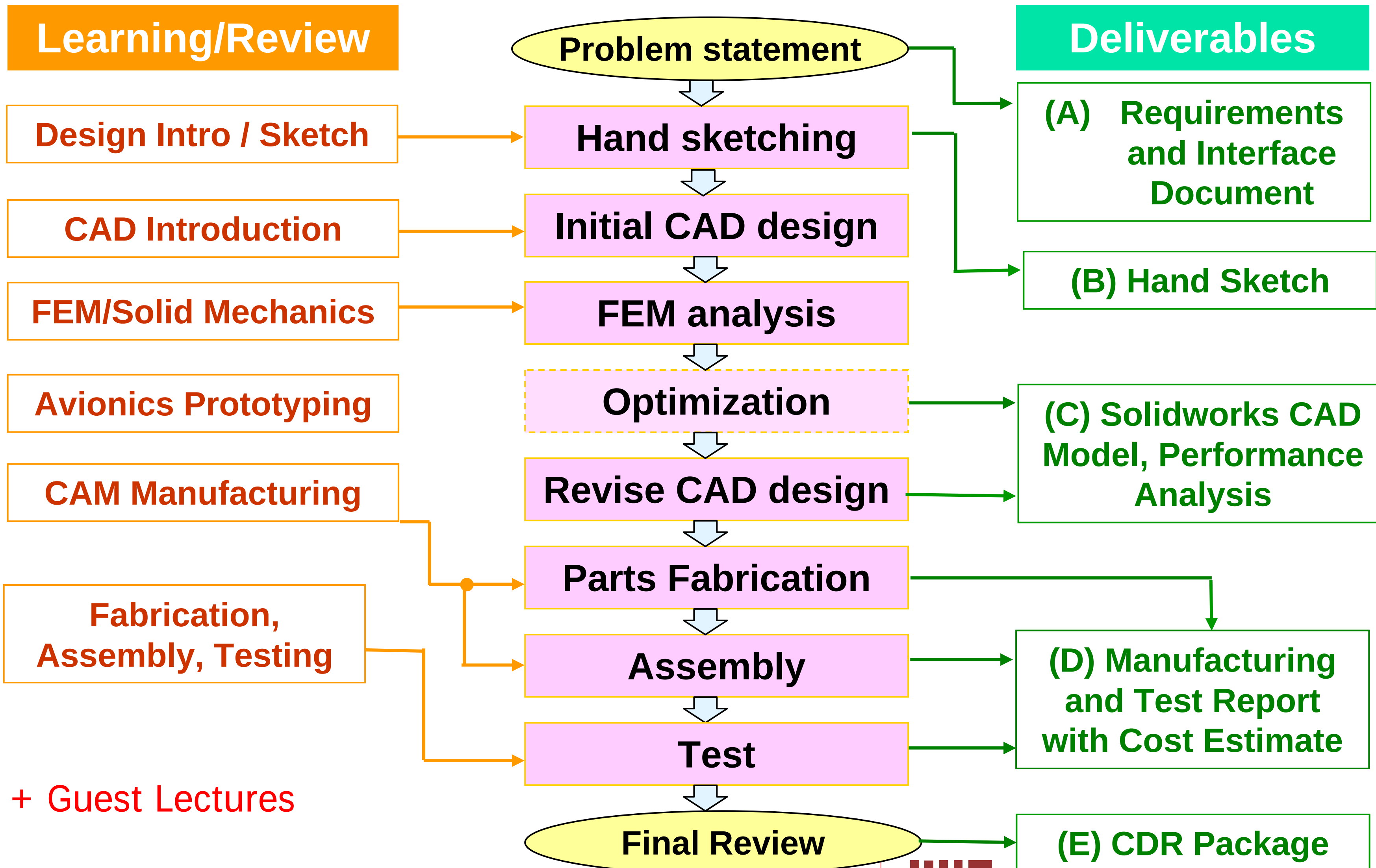
Plan for Today

- CAD Lecture (ca. 50 min)
  - CAD History, Background
  - Some theory on geometrical representation
- FEM Lecture (ca. 50 min)
  - Motivation for Structural Analysis
  - FEM Background
- Break
- Start creating your own CAD models (ca. 2 hrs)
  - Work in teams of two
  - Follow “User Manual” step-by-step, sample part
  - Then start on your own team projects
  - Use hand sketch (deliverable B) as starting point

Course Concept



Course Flow Diagram (2007)



16.810 What is CAD?

- Computer Aided Design (CAD)
  - A set of methods and tools to assist product designers in
    - Creating a geometrical representation of the artifacts they are designing
    - Dimensioning, Tolerancing
    - Configuration Management (Changes)
    - Archiving
    - Exchanging part and assembly information between teams, organizations
    - Feeding subsequent design steps
      - Analysis (CAE)
      - Manufacturing (CAM)
  - ...by means of a computer system.

16.810 Basic Elements of a CAD System

Input Devices

Keyboard
Mouse

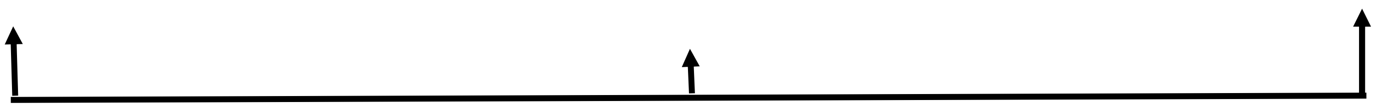
CAD keyboard
Templates
Space Ball

Main System

Computer
CAD Software
Database

Output Devices

Hard Disk
Network
Printer
Plotter



Human Designer

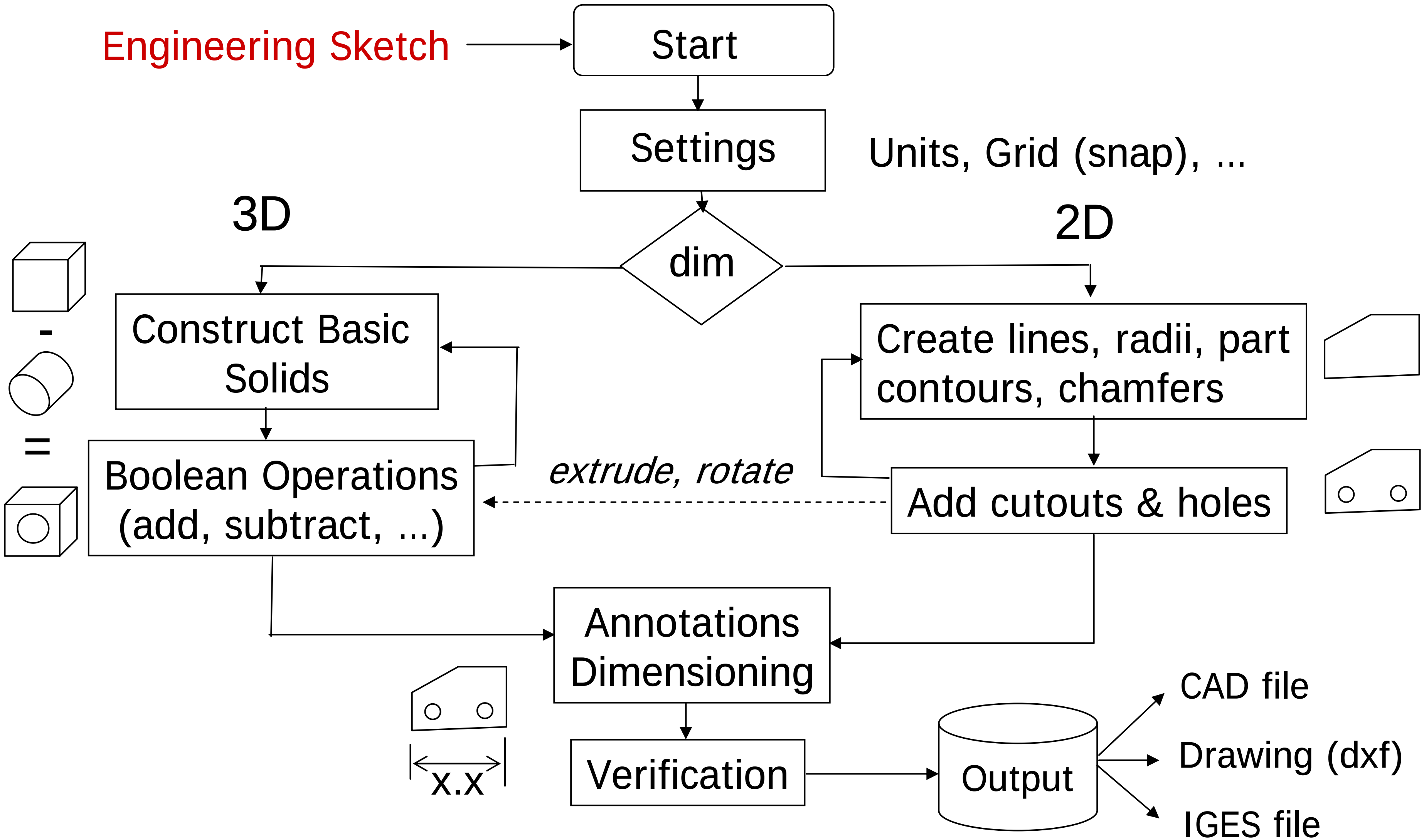
Brief History of CAD

- 1957 PRONTO (Dr. Hanratty) – first commercial numerical-control programming system
- 1960 SKETCHPAD (MIT Lincoln Labs)
- Early 1960's industrial developments
  - General Motors – DAC (Design Automated by Computer)
  - McDonnell Douglas – CADD
- Early technological developments
  - Vector-display technology
  - Light-pens for input
  - Patterns of lines rendering (first 2D only)
- 1967 Dr. Jason R Lemon founds SDRC in Cincinnati
- 1979 Boeing, General Electric and NIST develop IGES (Initial Graphic Exchange Standards), e.g. for transfer of NURBS curves
- Since 1981: numerous commercial programs
    - Source: http://mbinfo.mbdesign.net/CAD-History.htm

16.810 Major Benefits of CAD

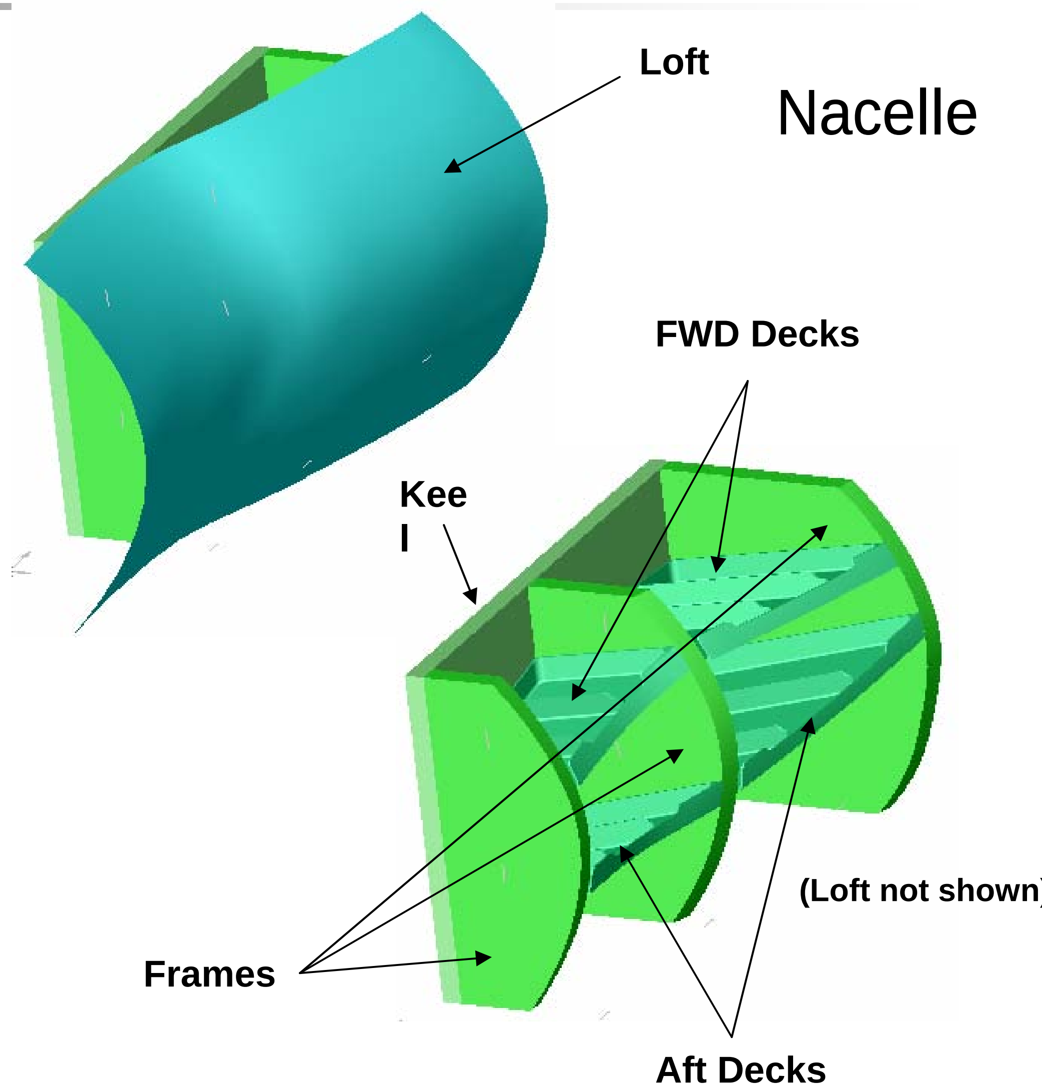
- Productivity (=Speed) Increase
  - Automation of repeated tasks
    - Doesn't necessarily increase creativity!
  - Insert standard parts (e.g. fasteners) from database
- Supports Changeability
  - Don't have to redo entire drawing with each change
    - EO – "Engineering Orders"
  - Keep track of previous design iterations
- Communication
  - With other teams/engineers, e.g. manufacturing, suppliers
  - With other applications (CAE/FEM, CAM)
  - Marketing, realistic product rendering
  - Accurate, high quality drawings
    - Caution: CAD Systems produce errors with hidden lines etc...
- Some limited Analysis
  - Mass Properties (Mass, Inertia)
  - Collisions between parts, clearances

Generic CAD Process



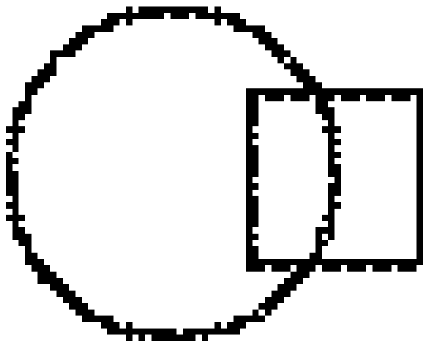
Example CAD A/C Assembly

- Boeing (sample) parts
  - A/C structural assembly
    - 2 decks
    - 3 frames
    - Keel
  - Loft included to show interface/stayout zone to A/C
  - All Boeing parts in Catia file format
    - Files imported into SolidWorks by converting to IGES format



Vector versus Raster Graphics

Raster Graphics

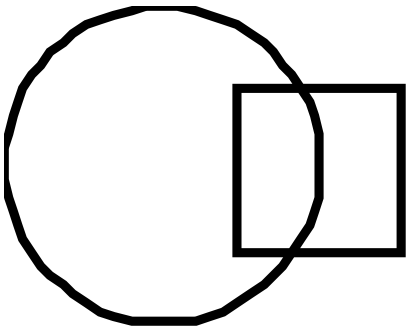


- Grid of pixels
  - No relationships between pixels
  - Resolution, e.g. 72 dpi (dots per inch)
  - Each pixel has color, e.g. 8-bit image has 256 colors

.bmp - raw data format


```
42 4D BC 02 00 00 00 00 00 00 3E 00 00 00 28 00 00 00 42 00 00 00 35 00 00 00 01
00 01 00 00 00 00 00 00 00 00 00 12 0B 00 00 12 0B 00 00 00 00 00 00 00 00 00 00
FF FF FF 00 00 00 00 00 00 00 15 FD 00 00 00 00 00 00 00 00 00 00 FF EF F8 00 00
00 00 00 00 00 00 01 D0 00 5C 00 00 00 00 00 00 00 00 0F 80 00 0F 80 00 00 00 00
00 00 00 1C 00 00 01 40 00 00 00 00 00 00 00 38 00 00 00 E0 00 00 00 00 00 00 00
70 00 00 00 70 00 00 00 00 00 00 00 E0 00 00 00 38 00 00 00 00 00 00 01 C0 00 00
00 1C 00 00 00 00 00 00 07 80 00 00 00 0E 00 00 00 00 00 00 07 00 00 00 00 07 00
00 00 00 00 00 0E 00 00 00 03 BB BB BB 80 00 00 00 1C 00 00 00 03 FF FF FF C0 00
00 00 18 00 00 00 03 00 C0 00 40 00 00 00 10 00 00 00 03 00 40 00 40 00 00 00 30
00 00 00 02 00 60 00 40 00 00 00 70 00 00 00 03 00 50 00 40 00 00 00 60 00 00 00
02 00 70 00 40 00 00 00 40 00 00 00 03 00 10 00 40 00 00 00 E0 00 00 00 03 00 30
00 40 00 00 00 40 00 00 00 03 00 10 00 40 00 00 00 C0 00 00 00 03 00 18 00 40 00
00 00 40 00 00 00 03 00 10 00 40 00 00 00 C0 00 00 00 02 00 18 00 40 00 00 00 C0
00 00 00 03 00 18 00 40 00 00 00 C0 00 00 00 02 00 08 00 40 00 00 00 C0 00 00 00
03 00 18 00 40 00 00 00 80 00 00 00 03 00 18 00 40 00 00 00 C0 00 00 00 03 00 10
00 40 00 00 00 80 00 00 00 03 00 18 00 40 00 00 00 40 00 00 00 03 00 10 00 40 00
00 00 C0 00 00 00 02 00 18 00 40 00 00 00 40 00 00 00 03 00 10 00 40 00 00 00 E0
00 00 00 02 00 38 00 40 00 00 00 40 00 00 00 03 00 10 00 40 00 00 00 60 00 00 00
03 00 30 00 40 00 00 00 70 00 00 00 03 00 70 00 40 00 00 00 30 00 00 00 03 00 60
00 40 00 00 00 10 00 00 00 03 77 77 77 40 00 00 00 18 00 00 00 03 FF FF FF C0 00
00 00 1C 00 00 00 00 01 C0 00 00 00 00 00 0E 00 00 00 00 03 80 00 00 00 00 00 07
00 00 00 00 07 00 00 00 00 00 00 03 00 00 00 00 0E 00 00 00 00 00 00 01 00 00 00
00 14 00 00 00 00 00 00 01 E0 00 00 00 38 00 00 00 00 00 00 00 70 00 00 00 70 00
00 00 00 00 00 00 38 00 00 00 E0 00 00 00 00 00 00 00 1C 00 00 01 C0 00 00 00 00
00 00 00 0F 80 00 0F 80 00 00 00 00 00 00 00 01 D0 00 5C 00 00 00 00 00 00 00 00
00 FF BB F8 00 00 00 00 00 00 00 00 00 17 FF 40 00 00 00 00 00 00 00 00 00
```

16.810 Vector Graphics

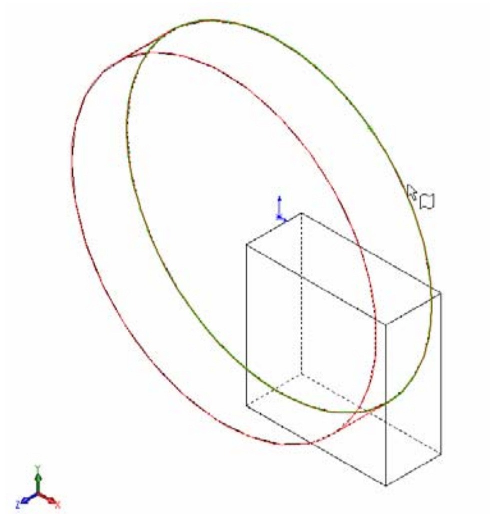


.emf format

CAD Systems use vector graphics

- Object Oriented
  - relationship between pixels captured
  - describes both (anchor/control) points and lines between them
  - Easier scaling & editing

Most common interface file: IGES



16.810 Major CAD Software Products

- AutoCAD (Autodesk) → mainly for PC
- Pro Engineer (PTC)
- SolidWorks (Dassault Systems)
- CATIA (IBM/Dassault Systems)
- Unigraphics (UGS)
- I-DEAS (SDRC)

Some CAD-Theory

Geometrical representation

(1) Parametric Curve Equation vs.

Nonparametric Curve Equation

(2) Various curves (*some mathematics !*)

- **Hermite Curve**
- **Bezier Curve**
- **B-Spline Curve**
- **NURBS (Nonuniform Rational B-Spline) Curves**

Applications: CAD, FEM, Design Optimization

Curve Equations

Two types of equations for curve representation

(1) Parametric equation

x, y, z coordinates are related by a parametric variable $(u \text{ or } \theta)$

(2) Nonparametric equation

x, y, z coordinates are related by a function

Example: Circle (2-D)

Parametric equation

$$x = R\cos\theta, \quad y = R\sin\theta \quad (0 \le \theta \le 2\pi)$$

Nonparametric equation

$$x^2 + y^2 - R^2 = 0$$ **(Implicit nonparametric form)**

$$y = \pm\sqrt{R^2 - x^2}$$ **(Explicit nonparametric form)**

Curve Equations

Two types of curve equations

(1) Parametric equation

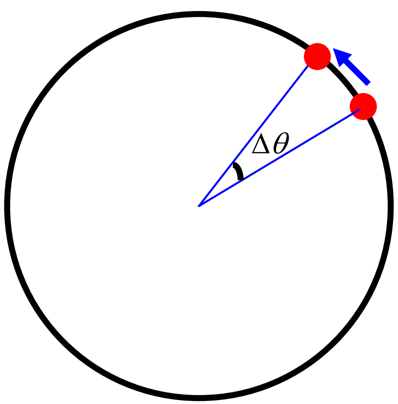
Point on 2-D curve: $\mathbf{p} = [x(u) \quad y(u)]$

Point on 3-D surface: $\mathbf{p} = [x(u) \quad y(u) \quad z(u)]$

u : parametric variable and independent variable

(2) Nonparametric equation $y = f(x)$: 2-D , $z = f(x, y)$: 3-D

Which is better for CAD/CAE? : Parametric equation



$$x = R\cos\theta, \quad y = R\sin\theta \quad (0 \le \theta \le 2\pi)$$

It also is good for calculating the points at a certain interval along a curve

$$x^2 + y^2 - R^2 = 0$$

$$y = \pm\sqrt{R^2 - x^2}$$

Parametric Equations –
Advantages over nonparametric forms

1. Parametric equations usually offer more degrees of freedom for controlling the shape of curves and surfaces than do nonparametric forms.

e.g. Cubic curve

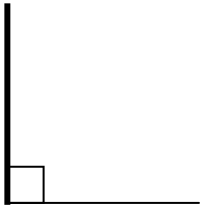
Parametric curve: $x = au^3 + bu^2 + cu + d$

$$y = eu^3 + fu^2 + gx + h$$

Nonparametric curve: $y = ax^3 + bx^2 + cx + d$

2. Parametric forms readily handle infinite slopes

$$\frac{dy}{dx} = \frac{dy/du}{dx/du} \quad \Rightarrow \; dx/du = 0 \;\; \text{indicates} \;\; dy/dx = \infty$$



3. Transformation can be performed directly on parametric equations

e.g. Translation in x-dir.

Parametric curve: $x = au^3 + bu^2 + cu + d + x_0$

$$y = eu^3 + fu^2 + gx + h$$

Nonparametric curve: $y = a(x - x_0)^3 + b(x - x_0)^2 + c(x - x_0) + d$

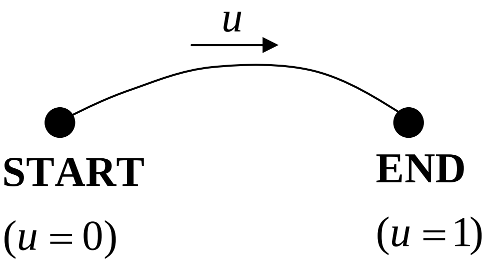
Hermite Curves

* Most of the equations for curves used in CAD software are of degree 3, because two curves of degree 3 guarantees 2nd derivative continuity at the connection point
→ The two curves appear to be one.

* Use of a higher degree causes small oscillations in curves and requires heavy computation.

* Simplest parametric equation of degree 3

$$\mathbf{P}(u) = [x(u) \;\; y(u) \;\; z(u)]$$
$$= \mathbf{a}_0 + \mathbf{a}_1 u + \mathbf{a}_2 u^2 + \mathbf{a}_3 u^3 \qquad (0 \le u \le 1)$$



$\mathbf{a}_0, \; \mathbf{a}_1, \; \mathbf{a}_2, \; \mathbf{a}_3$: Algebraic vector coefficients

The curve's shape change cannot be intuitively anticipated from changes in these values

Hermite Curves

$$\mathbf{P}(u) = \mathbf{a}_0 + \mathbf{a}_1 u + \mathbf{a}_2 u^2 + \mathbf{a}_3 u^3 \qquad (0 \le u \le 1)$$

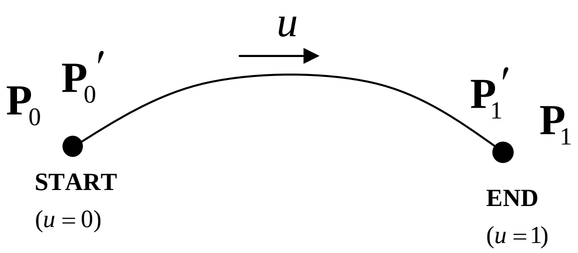
Instead of algebraic coefficients, let's use the position vectors and the tangent vectors at the two end points!

Position vector at starting point: $\mathbf{P}_0 = \mathbf{P}(0) = \mathbf{a}_0$

Position vector at end point: $\mathbf{P}_1 = \mathbf{P}(1) = \mathbf{a}_0 + \mathbf{a}_1 + \mathbf{a}_2 + \mathbf{a}_3$

Tangent vector at starting point: $\mathbf{P}_0' = \mathbf{P}'(0) = \mathbf{a}_1$

Tangent vector at end point: $\mathbf{P}_1' = \mathbf{P}'(1) = \mathbf{a}_1 + 2\mathbf{a}_2 + 3\mathbf{a}_3$



Blending functions

$$\mathbf{P}(u) = [1 - 3u^2 + 2u^3 \quad 3u^2 - 2u^3 \quad u - 2u^2 + u^3 \quad -u^2 + u^3] \begin{bmatrix} \mathbf{P}_0 \\ \mathbf{P}_1 \\ \mathbf{P}_0' \\ \mathbf{P}_1' \end{bmatrix}$$

: Hermit curve

No algebraic coefficients

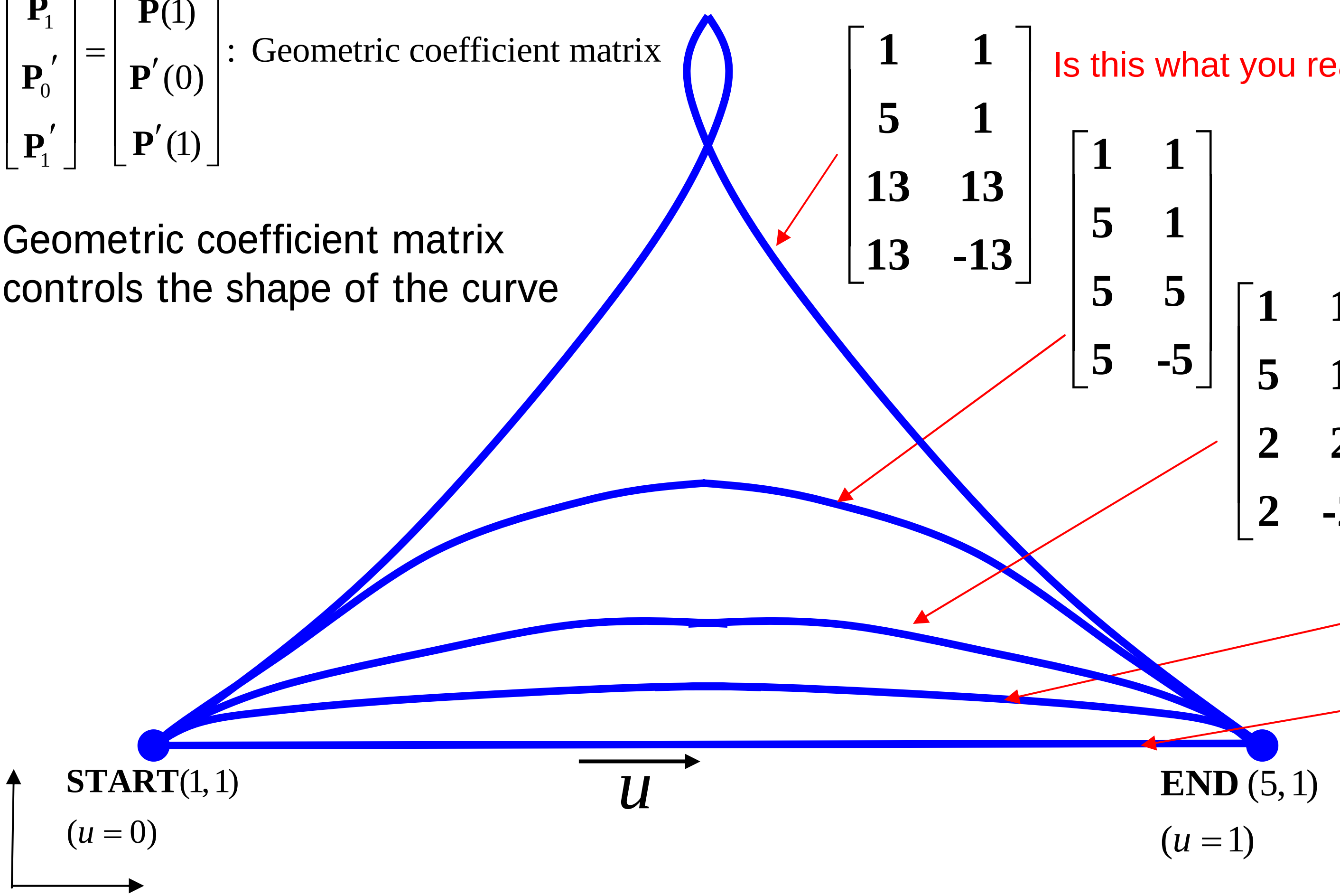
$\mathbf{P}_0, \mathbf{P}_0', \mathbf{P}_1, \mathbf{P}_1'$: Geometric coefficients

➡ The curve's shape change can be intuitively anticipated from changes in these values

Effect of tangent vectors on the curve's shape

$$\begin{bmatrix} \mathbf{P}_0 \\ \mathbf{P}_1 \\ \mathbf{P}_0' \\ \mathbf{P}_1' \end{bmatrix} = \begin{bmatrix} \mathbf{P}(0) \\ \mathbf{P}(1) \\ \mathbf{P}'(0) \\ \mathbf{P}'(1) \end{bmatrix} : \text{ Geometric coefficient matrix}$$

Geometric coefficient matrix controls the shape of the curve



$$\begin{bmatrix} 1 & 1 \\ 5 & 1 \\ 13 & 13 \\ 13 & -13 \end{bmatrix} \quad \begin{bmatrix} 1 & 1 \\ 5 & 1 \\ 5 & 5 \\ 5 & -5 \end{bmatrix} \quad \begin{bmatrix} 1 & 1 \\ 5 & 1 \\ 2 & 2 \\ 2 & -2 \end{bmatrix} \quad \begin{bmatrix} 1 & 1 \\ 5 & 1 \\ 1 & 1 \\ 1 & -1 \end{bmatrix} \quad \begin{bmatrix} 1 & 1 \\ 5 & 1 \\ 4 & 0 \\ 4 & 0 \end{bmatrix}$$

$$\frac{dy}{dx} = \frac{dy/du}{dx/du} = \frac{0}{4} = 0$$

Bezier Curve

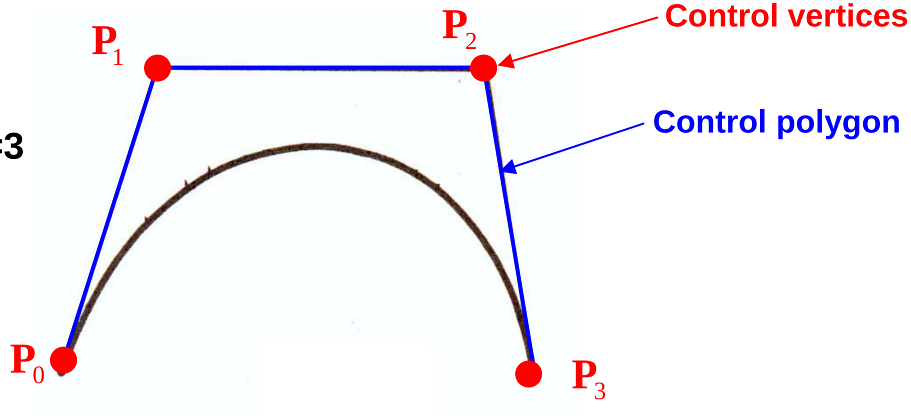
* In case of Hermite curve, it is not easy to predict curve shape caused by changes in the magnitude of the tangent vectors, $\mathbf{P}_0'$ and $\mathbf{P}_1'$

* Bezier Curve can control curve shape more easily using several control points (Bezier 1960)

$$\mathbf{P}(u) = \sum_{i=0}^{n} \binom{n}{i} u^i (1-u)^{n-i} \mathbf{P}_i \quad , \qquad \text{where} \quad \binom{n}{i} = \frac{n!}{i!(n-i)!}$$

$\mathbf{P}_i$: Position vector of the *i* th vertex (control vertices)

n=3



* Number of vertices: n+1 (No of control points)

* Number of segments: n

* Order of the curve: n

* The order of Bezier curve is determined by the number of control points.

n control points ⇨ Order of Bezier curve: n-1

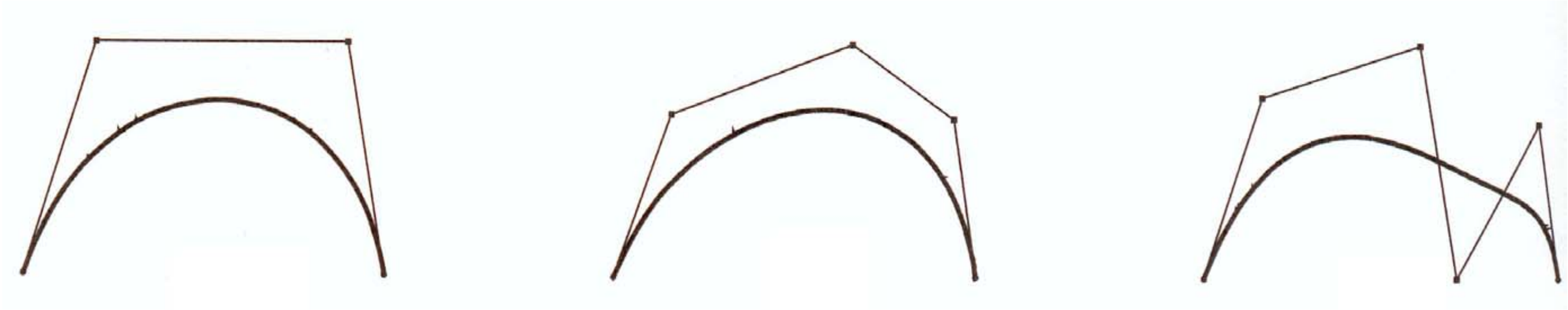
Bezier Curve

Properties

- The curve passes through the first and last vertex of the polygon.

-The tangent vector at the starting point of the curve has the same direction as the first segment of the polygon.

- The *n*th derivative of the curve at the starting or ending point is determined by the first or last (*n*+1) vertices.



Two Drawbacks of Bezier curve

(1) For complicated shape representation, higher degree Bezier curves are needed.

→ Oscillation in curve occurs, and computational burden increases.

(2) Any one control point of the curve affects the shape of the entire curve.

→ Modifying the shape of a curve locally is difficult.

(Global modification property)

Desirable properties :

1. Ability to represent complicated shape with low order of the curve

2. Ability to modify a curve's shape locally

⇨ B-spline curve!

B-Spline Curve

* Developed by Cox and Boor (1972)

$$\mathbf{P}(u) = \sum_{i=0}^{n} N_{i,k}(u)\mathbf{P}_i$$

where

$\mathbf{P}_i$: Position vector of the ith control point

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

$$N_{i,1}(u) = \begin{cases} 1 & t_i \le u \le t_{i+1} \\ 0 & \text{otherwise} \end{cases}$$

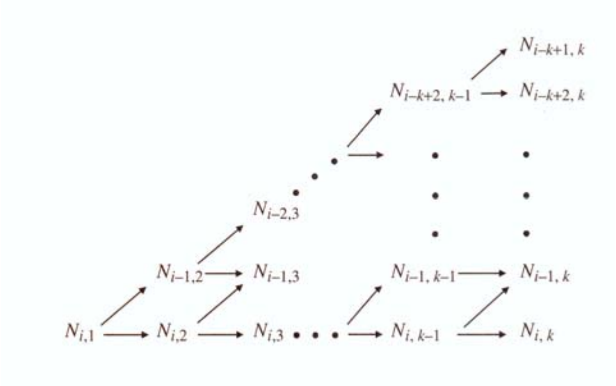
$$t_i = \begin{cases} 0 & 0 \le i < k \\ i - k + 1 & k \le i \le n \\ n - k + 2 & n < i \le n + k \end{cases}$$

(Nonperiodic knots)

k: order of the B-spline curve

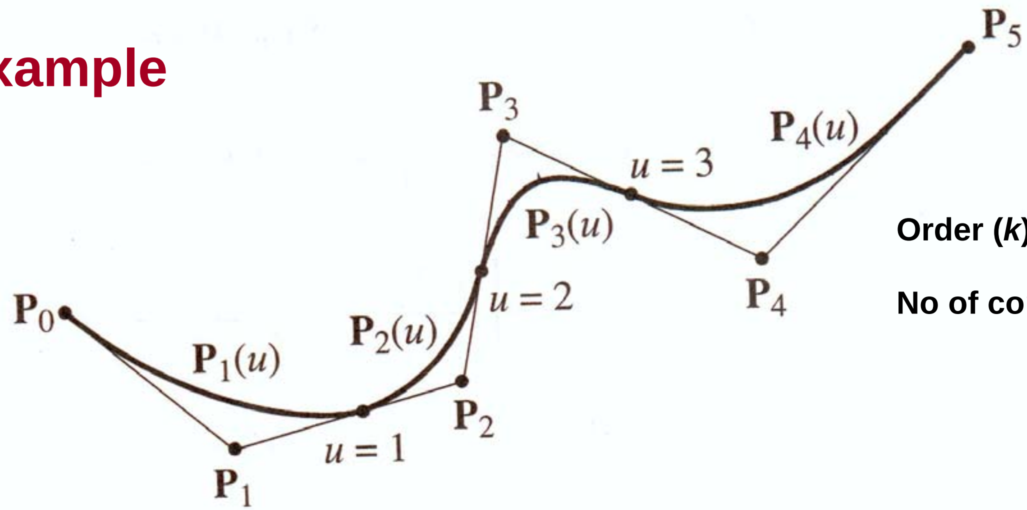
n+1: number of control points

The order of curve is independent of the number of control points!



B-Spline Curve

Example



Order (*k*) = 3 (first derivatives are continuous)

No of control points (*n+1*) = 6

Advantages

(1) The order of the curve is independent of the number of control points (contrary to Bezier curves)

- User can select the curve's order and number of control points separately.
- It can represent very complicated shape with low order

(2) Modifying the shape of a curve locally is easy. (contrary to Bezier curve)

- Each curve segment is affected by *k (order)* control points. (local modification property)

NURBS (Nonuniform Rational B-Spline) Curve

$$\mathbf{P}(u) = \frac{\sum_{i=0}^{n} h_i \mathbf{P}_i N_{i,k}(u)}{\sum_{i=0}^{n} h_i N_{i,k}(u)}$$

$$\left(\text{B-spline}: \ \mathbf{P}(u) = \sum_{i=0}^{n} \mathbf{P}_i N_{i,k}(u) \right)$$

$\mathbf{P}_i$: Position vector of the ith control point

h_i : Homogeneous coordinate

*** If all the homogeneous coordinates (h_i) are 1, the denominator becomes 1**

If $h_i = 0 \ \forall i$, then $\sum_{i=0}^{n} h_i N_{i,k}(u) = 1$.

*** B-spline curve is a special case of NURBS.**

*** Bezier curve is a special case of B-spline curve.**

Advantages of NURBS Curve over B-Spline Curve

(1) More versatile modification capacity

- Homogeneous coordinate h_i, which B-spline does not have, can change.
- Increasing h_i of a control point → Drawing the curve toward the control point.

(2) NURBS can exactly represent the conic curves - circles, ellipses, parabolas, and hyperbolas (B-spline can only approximate these curves)

(3) Curves, such as conic curves, Bezier curves, and B-spline curves can be converted to their corresponding NURBS representations.

Summary

(1) Parametric Equation vs. Nonparametric Equation

(2) Various curves

- **Hermite Curve**
- **Bezier Curve**
- **B-Spline Curve**
- **NURBS (Nonuniform Rational B-Spline) Curve**

(3) Surfaces

- **Bilinear surface**
- **Bicubic surface**
- **Bezier surface**
- **B-Spline surface**
- **NURBS surface**

SolidWorks

- SolidWorks
  - Most popular CAD system in education
  - Will be used for this project
  - Do Self-Introduction via 16.810 User Manual
  - See also
    - http://www.solidworks.com (Student Section)