16.810 (16.682) Engineering Design and Rapid Prototyping

Lecture 3

Computer Aided Design (CAD)

Instructor(s)

Prof. Olivier de Weck
deweck@mit.edu

Dr. Il Yong Kim
kiy@mit.edu

January 9, 2004
Plan for Today

- CAD Lecture (ca. 50 min)
  - CAD History, Background
  - Some theory of geometrical representation
- SolidWorks Introduction (ca. 40 min)
  - Led by Bill Nadir (TA)
  - Follow along step-by-step
- Create CAD model of your part (ca. 90 min)
  - Work in teams of two
  - Use hand sketch as starting point
Course Concept

Phase 1
- Problem statement
- Sketch by hand
- CAD
- CAE
- Rapid Prototyping / Validation
  - Manufacturing / Test

Phase 2
- Design Optimization
- Optimum solution
- Rapid Prototyping / Validation
  - Manufacturing / Test
Course Flow Diagram

Learning/Review
- Design Intro
- CAD/CAM/CAE Intro
- FEM/Solid Mechanics Overview
- Manufacturing Training
- Structural Test “Training”
- Design Optimization

Problem statement
- Hand sketching
- CAD design
- FEM analysis
- Produce Part 1
- Test
- Optimization
- Produce Part 2
- Test
- Final Review

Deliverables
- Design Sketch v1
- Drawing v1
- Analysis output v1
- Part v1
- Experiment data v1
- Design/Analysis output v2
- Part v2
- Experiment data v2

Due today
- Design Sketch v1
- Drawing v1
- Analysis output v1
- Part v1
- Experiment data v1
- Design/Analysis output v2
- Part v2
- Experiment data v2

Monday
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.
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

Ref: menzelus.com
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
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

1. **Engineering Sketch**
2. **Start**
3. **Settings**
   - Units, Grid (snap), ...

3D: **Construct Basic Solids**
   - Boolean Operations (add, subtract, ...)
   - Annotations Dimensioning
5. **Verification**
6. **Output**
   - CAD file
   - Drawing (dxf)
   - IGES file

2D: **Create lines, radii, part contours, chamfers**
   - Add cutouts & holes
   - Extrude, rotate
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
16.810  

Drawing Tree

L0: Top Kit Collector

L1: Avionics Sub Kit
  - Modification Kit
  - E/M\*22 PLC

L1: Elec Harness Sub Kit

L1: Airframe Sub Kit

L2: Turret

Avionics

L2: Cockpit

Avionics

L2: Nose Floor

L2: Transition

L3: Adds/Removes Hardware & Details

© Sikorsky
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
Vector Graphics

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

.CAD Systems use vector graphics

Most common interface file: IGES

.emf format
Major CAD Software Products

- AutoCAD (Autodesk) \( \rightarrow \) 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 \((u\ or\ \theta)\)

(2) Nonparametric equation
x, y, z coordinates are related by a function

Example: Circle (2-D)

Parametric equation
\[ x = R \cos \theta, \quad y = R \sin \theta \quad (0 \leq \theta \leq 2\pi) \]

Nonparametric equation
\[ x^2 + y^2 - R^2 = 0 \quad \text{(Implicit nonparametric form)} \]
\[ y = \pm \sqrt{R^2 - x^2} \quad \text{(Explicit nonparametric form)} \]
Curve Equations

Two types of curve equations

(1) Parametric equation

Point on 2-D curve: \( p = [x(u) \quad y(u)] \)
Point on 3-D surface: \( p = [x(u) \quad y(u) \quad z(u)] \)

\( u \): 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

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

\[
\begin{align*}
  x &= R \cos \theta, \\
  y &= R \sin \theta \quad (0 \leq \theta \leq 2\pi) \\
  x^2 + y^2 - R^2 &= 0 \\
  y &= \pm \sqrt{R^2 - x^2}
\end{align*}
\]
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
   
   \[ \frac{dy}{dx} = \frac{dy/du}{dx/du} \quad \Rightarrow \quad \frac{dx}{du} = 0 \quad \text{indicates} \quad \frac{dy}{dx} = \infty \]

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:**
   \[ y = a(x - x_0)^3 + b(x - x_0)^2 + c(x - x_0)^2 + d \]
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 one.

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

* Simplest parametric equation of degree 3

\[
\mathbf{P}(u) = [x(u) \ y(u) \ z(u)] = a_0 + a_1u + a_2u^2 + a_3u^3 \quad (0 \leq u \leq 1)
\]

\(a_0, a_1, a_2, a_3\): Algebraic vector coefficients

→ The curve’s shape change cannot be intuitively anticipated from changes in these values
Hermite Curves

\[ P(u) = a_0 + a_1u + a_2u^2 + a_3u^3 \quad (0 \leq u \leq 1) \]

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

Position vector at starting point: \( P_0 = P(0) = a_0 \)

Position vector at end point: \( P_1 = P(1) = a_0 + a_1 + a_2 + a_3 \)

Tangent vector at starting point: \( P_0' = P'(0) = a_1 \)

Tangent vector at end point: \( P_1' = P'(1) = a_1 + 2a_2 + 3a_3 \)

\[
P(u) = \begin{bmatrix} 1 - 3u^2 + 2u^3 & 3u^2 - 2u^3 & u - 2u^2 + u^3 & -u^2 + u^3 \end{bmatrix} \begin{bmatrix} P_0 \\ P_1 \\ P_0' \\ P_1' \end{bmatrix}
\]

No algebraic coefficients

\( P_0, P_0', P_1, P_1' \): 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

\[
\begin{bmatrix}
P_0 \\
P_1 \\
P'_0 \\
P'_1
\end{bmatrix} = \begin{bmatrix}
P(0) \\
P(1) \\
P'(0) \\
P'(1)
\end{bmatrix} : \text{Geometric coefficient matrix}
\]

Geometric coefficient matrix controls the shape of the curve

Is this what you really wanted?

\[
\begin{bmatrix}
1 & 1 \\
5 & 1 \\
13 & 13 \\
13 & -13
\end{bmatrix}
\]

\[
\begin{bmatrix}
1 & 1 \\
5 & 1 \\
5 & 5 \\
5 & -5
\end{bmatrix}
\]

\[
\begin{bmatrix}
1 & 1 \\
2 & 2 \\
2 & -2
\end{bmatrix}
\]

\[
\begin{bmatrix}
1 & 1 \\
1 & 1 \\
1 & -1
\end{bmatrix}
\]

\[
\begin{bmatrix}
1 & 1 \\
5 & 1 \\
4 & 0 \\
4 & 0
\end{bmatrix}
\]

\[
\frac{dy}{dx} = \frac{dy}{du} = 0 \quad \frac{dx}{du} = 0
\]
* In case of Hermite curve, it is not easy to predict curve shape according to changes in magnitude of the tangent vectors, $P_0'$ and $P_1'$

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

$$P(u) = \sum_{i=0}^{n} \binom{n}{i} u^i (1-u)^{n-i} P_i,$$

where \( \binom{n}{i} = \frac{n!}{i!(n-i)!} \)

$P_i$: Position vector of the $i$ th vertex (control vertices)

* 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.
Beziers 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

\[ P(u) = \sum_{i=0}^{n} N_{i,k}(u) P_i \]

where

- \( P_i \): Position vector of the \( i \)th control point

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

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

- \( k \): order of the B-spline curve
- \( n+1 \): number of control points

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

* Developed by Cox and Boor (1972)
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

\[ P(u) = \frac{\sum_{i=0}^{n} h_i P_i N_{i,k}(u)}{\sum_{i=0}^{n} h_i N_{i,k}(u)} \]

\[ \left( \text{B-spline: } P(u) = \sum_{i=0}^{n} P_i N_{i,k}(u) \right) \]

\[ P_i : \text{Position vector of the } i\text{th control point} \]
\[ h_i : \text{Homogeneous coordinate} \]

* If all the homogeneous coordinates \((h_i)\) are 1, the denominator becomes 1
  
  If \(h_i = 0 \quad \forall i\), 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 $\rightarrow$ 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
Flat surface

Not separate parts (the surface at the position must be flat)
Design Freedom

Both the flat surface and holes move together along the design freedom line

Design freedom: ± 0.800
Design freedom: ± 0.100

45°
Displacement

Horizontal displacement or rotation is fine!
Linear vs. Nonlinear deformation

Order of applying loads

- For linear deformation, it does not matter.
- For nonlinear deformation (e.g., buckling), it is important.
# IAP 2004 Schedule

<table>
<thead>
<tr>
<th>Week</th>
<th>Monday</th>
<th>Wednesday</th>
<th>Friday</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Lecture</td>
<td>L1 – Introduction (de Weck)</td>
<td>L2 – Hand Sketching (Wallace)</td>
</tr>
<tr>
<td></td>
<td>Hands-on activities</td>
<td>Tour - Design studio - Machine shop - Testing area</td>
<td>Sketch Initial design</td>
</tr>
<tr>
<td>2</td>
<td>Lecture</td>
<td>L4 – Introduction to CAE (Kim)</td>
<td>L5 – Introduction to CAM (Kim)</td>
</tr>
<tr>
<td></td>
<td>Hands-on activities</td>
<td>FEM Analysis (Cosmos)</td>
<td>Water Jet Intro machine shop Omax (Weiner, Nadir)</td>
</tr>
<tr>
<td>3</td>
<td>Lecture</td>
<td>Martin Luther King Jr. Holiday – no class</td>
<td>L7 – Structural Testing (Kim, de Weck)</td>
</tr>
<tr>
<td></td>
<td>Hands-on activities</td>
<td></td>
<td>Test part ver. 1 (Kane)</td>
</tr>
<tr>
<td>4</td>
<td>Lecture</td>
<td>Carry out design optimization</td>
<td>Manufacture part ver. 2 Test part ver. 2</td>
</tr>
</tbody>
</table>
**SolidWorks Introduction**

- **SolidWorks**
  - Most popular CAD system in education
  - Will be used for this project
  - 40 Minute Introduction by Bill Nadir (TA)
    - [http://www.solidworks.com](http://www.solidworks.com) (Student Section)