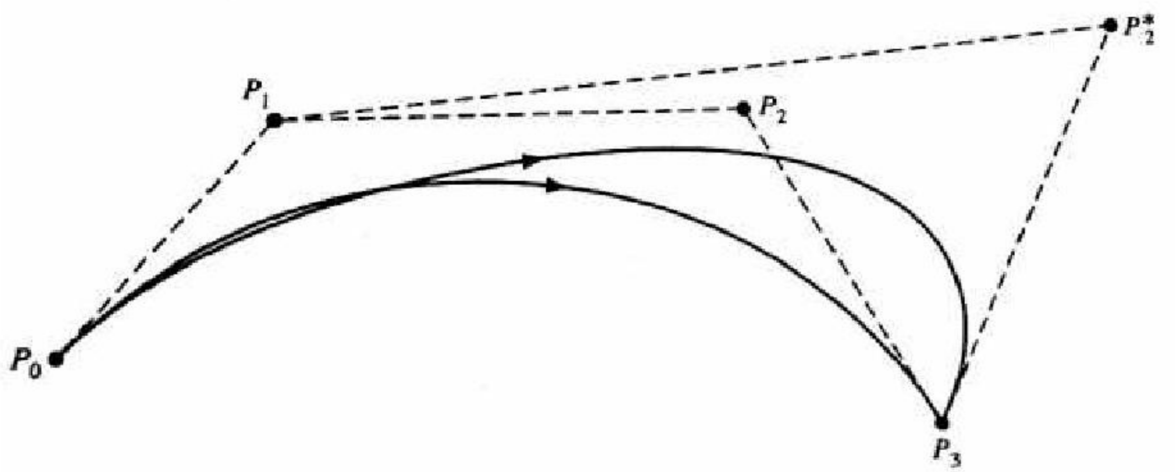
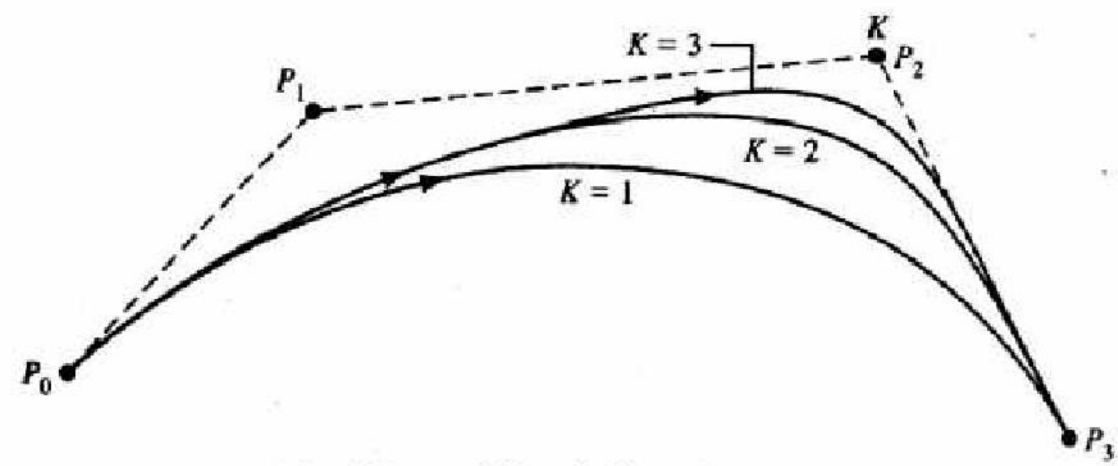
UNIT-II- GEOMETRIC MODELING

CHARACTERISTICS OF BEZIER CURVE

- The curve interpolates the first and last control points.
- The curve is tangent to the first and last segments of characteristics polygon.
- The curve shape can be modified by either changing one or more vertices of its polygon or by keeping the polygon fixed and specifying multiple coincident points at a vertex as shown in figure.



(*a*) Changing a vertex



(*b*) Specifying multiple coincident points at a vertex

PRE-REQUISITE DISCUSSION

CURVE REPRESENTATIO N

(1) Parametric equation x, y, z coordinates are related by a parametric variable (u or θ)
(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

$$x^2 + y^2 - R^2 = 0 \quad \text{(Implicit nonparametric form)}$$

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

TYPES OF CURVES USED IN GEOMETRIC MODELLING

- Hermite curves
- Bezeir curves
- B-spline curves
  - NURBS

curves

HERMITE CURVES

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

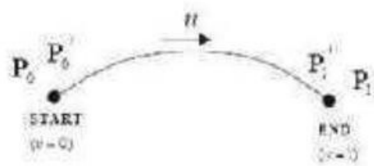
Instead of algebraic coefficients, let's use the position vectors and the tangent vectors at the two end points!

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

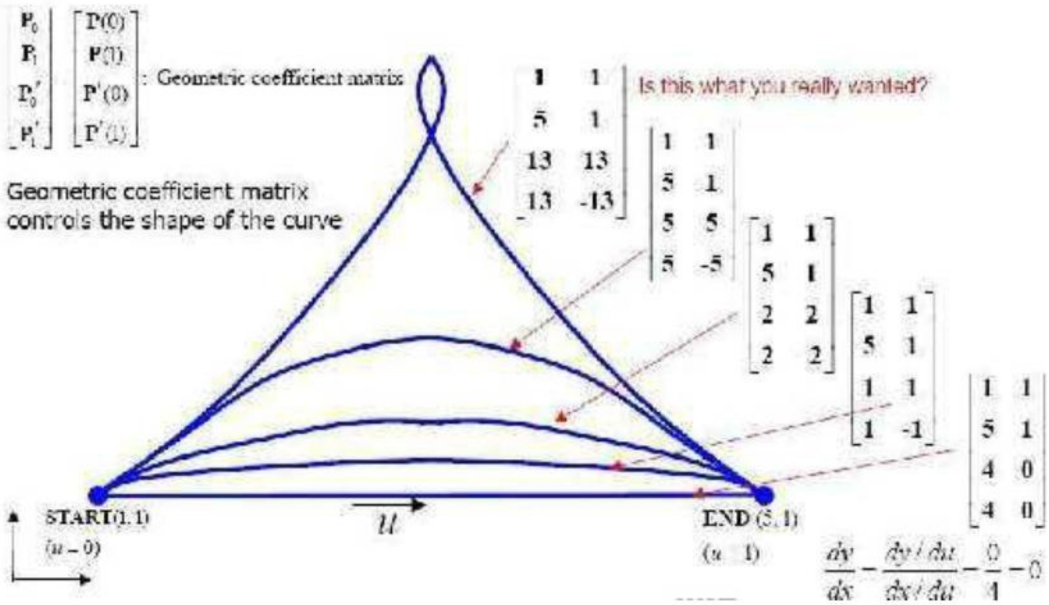
$$\mathbf{P}(u) = \begin{bmatrix} 1-3u^2+2u^3 & 3u^2-2u^3 & u-2u^2+u^3 & -u^2+u^3 \end{bmatrix} \begin{bmatrix} \mathbf{P}_0 \\ \mathbf{P}_1 \\ \mathbf{P}_0' \\ \mathbf{P}_1' \end{bmatrix} \quad \text{: Hermit curve}$$

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 vector on t he curve's shape



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

Two Drawbacks of Bezier Curves

(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.

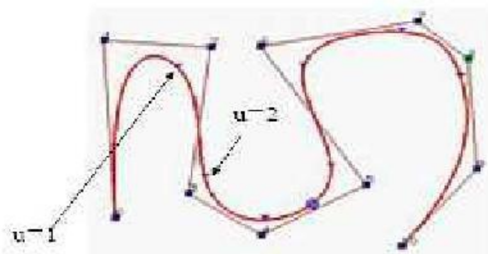
(Global modification property)

Desirable properties :

1. Ability to represent complicated shape with low order of the curve
2. Ability to modify a curve's shape locally

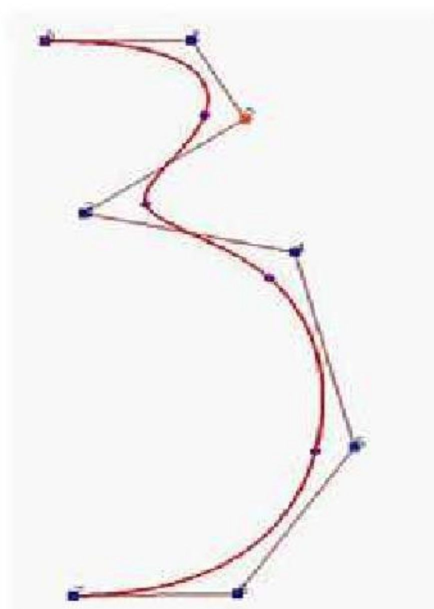
Properties of B-Spline

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

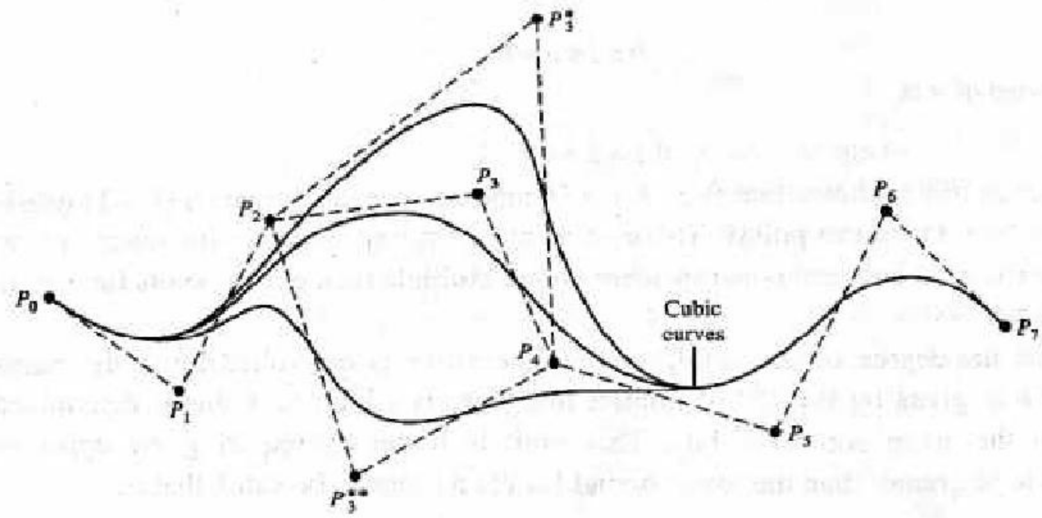


B-Spline

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



NURBS curve



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

$\mathbf{P}_i$: 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 $\sum_{i=0}^{n} h_i N_{i,3}(u) = 1$.

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

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

Advantages of B-spline curves and NURBS 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)

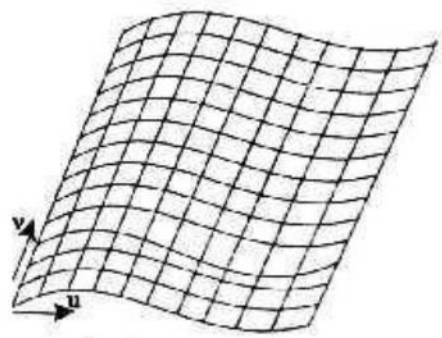
(3) Curves, such as conic curves, Bezier curves, and B-spline curves can be converted to their corresponding NURBS representations.

TECHNIQUES IN SURFACE MODELLING

i. Surface Patch
ii. Coons Patch
iii. Bicubic Patch
iv. Be'zier Surface
v. B-Spline Surface

i. Surface Patch

The patch is the fundamental building block for surfaces. The two variables *u* and *v* vary across the patch; the patch may be termed *biparametric.* The parametric variables often lie in the range 0 to 1. Fixing the value of one of the parametric variables results in a curve on the patch in terms of the other variable (*Isoperimetric curve*). Figure shows a surface with curves at intervals of *u* and *v* of 0 : 1.



Surface patch.

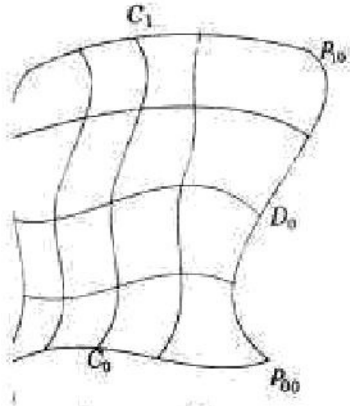
ii. Coons Patch

The sculptured surface often involve interpolation across an intersecting mesh of curves that in effect comprise a rectangular grid of patches, each bounded by four boundary curves. The linearly blended coons patch is the simplest for interpolating between such boundary curves. This patch definition technique blends for four boundary curves $C_i(u)$ and $D_j(v)$ and the corner points p_{ij} of the patch with the linear blending functions,

$$f(t) = 1 - t$$
$$g(t) = t$$

using the expression

$$\vec{p}(u, v) = \vec{C}_0(u)\, f(v) + \vec{C}_1(u)\, g(v) + \vec{D}_0(v)\, f(u) + \vec{D}_1(v)\, g(u)$$
$$- \vec{p}_{00}\, f(u)\, f(v) - \vec{p}_{01}\, f(u)\, g(u) - \vec{p}_{10}\, g(u)\, f(v) - \vec{p}_{11}\, g(u)\, g(v)$$



Coons patch

iii. Bicubic Patch

The bi-cubic patch is used for surface descriptions defined in terms of point and tangent vector information. The general form of the expressions for a bi-cubic patch is given by:

$$\vec{p}(u, v) = \sum_{i=0}^{3} \sum_{j=0}^{3} \vec{k_{ij}}\, u^i v^j$$

This is a vector equation with 16 unknown parameters *kij* which can be found by Lagrange interpolation through 4 x 4 grid.

iv. Be'zier Surface

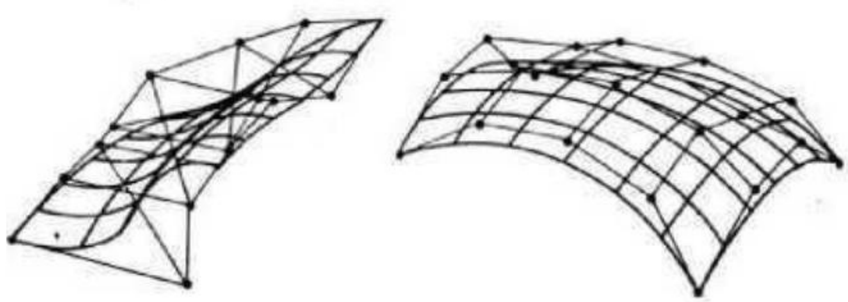
- The Be'zier surface formulation use a characteristic polygon
- Points the Bezier surface are given by

$$\vec{p}(u, v) = \sum_{i=0}^{m} \sum_{j=0}^{n} \vec{B}_{i,m}(u)\, \vec{B}_{j,n}\, \vec{p}_{ij};\ u, v \in [0, 1]$$

Where,

$\vec{p}_{ij}$ - Vertices of the characteristic polygon

$\vec{B}_{j,n}$, $\vec{B}_{i,m}$ - Blending functions



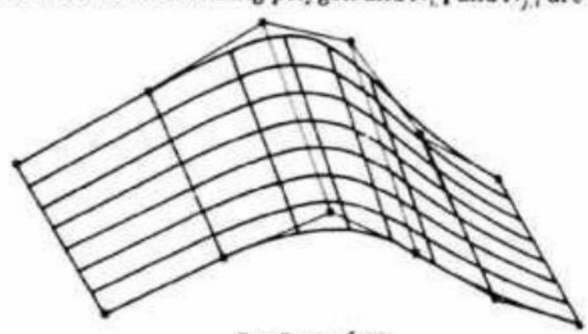
v. B-Spline Surfaces

- The B-spline surface approximates a characteristics polygon as shown and passes through the corner points of the polygon, where its edges are tangential to the edges of the polygon
- This may not happen when the control polygon is closed
- A control point of the surface influences the surface only over a limited portion of the parametric space of variables u and v.

The expression for the B-spline surfaces is given by

$$\vec{p}(u,v) = \sum_{i=0}^{m} \sum_{j=0}^{n} N_{i,k}(u) N_{j,l}(v) \vec{p_{ij}}$$

p_{ij} are the vertices of the defining polygon and $N_{i,k}$ and $N_{j,l}$ are blending functions



B-spline surface

B-spline surface for a closed control polygon

GEOMETRIC MODELLING

Geometric modeling is the starting point of the product design and manufacture process. Functions of Geometric Modeling are:

Design Analysis

Evaluation of area, volume, mass and inertia properties
Interference checking in assemblies
Analysis of tolerance build-up in assemblies
Kinematic analysis of mechanisms and robots
Automatic mesh generation for finite element analysis

Drafting

Automatic planar cross-sectioning
Automatic hidden lines and surface removal
Automatic production of shaded images
Automatic dimensioning
Automatic creation of exploded views of assemblies

Manufacturing

Parts classification
Process planning
NC data generation and verification
Robot program generation

Production Engineering

Bill of materials
Material requirement
Manufacturing resource requirement
Scheduling

Inspection and quality control

Program generation for inspection machines
Comparison of produced parts with design

PROPERTIES OF A GEOMETRIC MODELING SYSTEM

The geometric model must stay invariant with regard to its location and orientation The solid must have an interior and must not have isolated parts

The solid must be finite and occupy only a finite shape

The application of a transformation or Boolean operation must produce another solid The solid must have a finite number of surfaces which can be described

The boundary of the solid must not be ambiguous

WIRE FRAME MODELING

It uses networks of interconnected lines (wires) to represent the edges of the physical objects being modeled

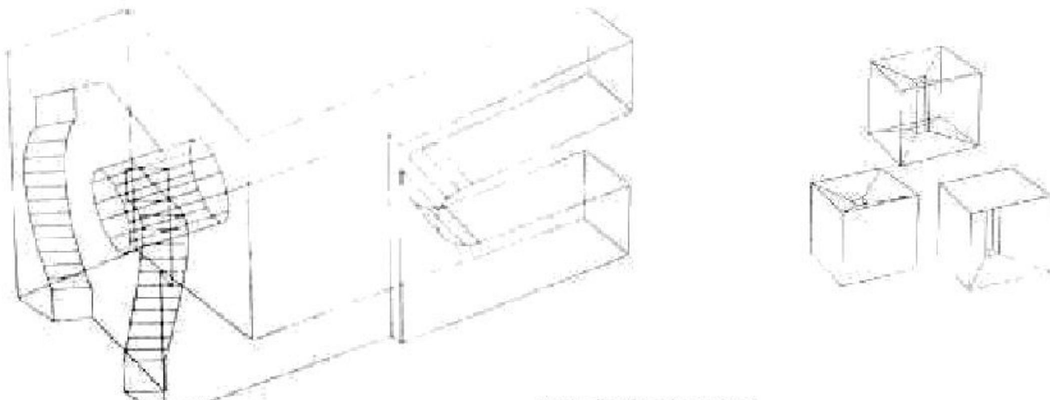
Also called _Edge-vertex‘ or _stick- figure‘ models

Two types of wire frame modeling:

1. 2 ½ - D modeling
2. 3 – D modeling

3-D Wire frame models: These are

Simple and easy to create, and they require relatively little computer time and memory; however they do not give a complete description of the part. They co ntain little information about the surface and volume of the part and cannot distinguish the inside from the outside of part surfaces. They are visually ambiguous as the model can be interpreted in many different ways because in many wire frame models hidden lines cannot be removed. Section property and mass calculations are impossible, since the object has no faces attached to it. It has limited values a basis for manufacture and analysis



3-D wireframe model

TECHNIQUES IN SURFACE MODELLING The various methods for representing the solids are:

1. Half-space method
2. Boundary representation method (B-rep)
3. Constructive solid geometry (CSG and C-rep)
4. Sweep representation
5. Analytical solid modeling (ASM)
6. Primitive instancing
7. Spatial partitioning representation
   a. Cell decomposition
   b. Spatial occupancy enumeration

Boundary representation method (B-rep)

- In solid modeling and computer-aided design, boundary representation often abbreviated as B-rep or BREP—is a method for representing shapes using the limits.

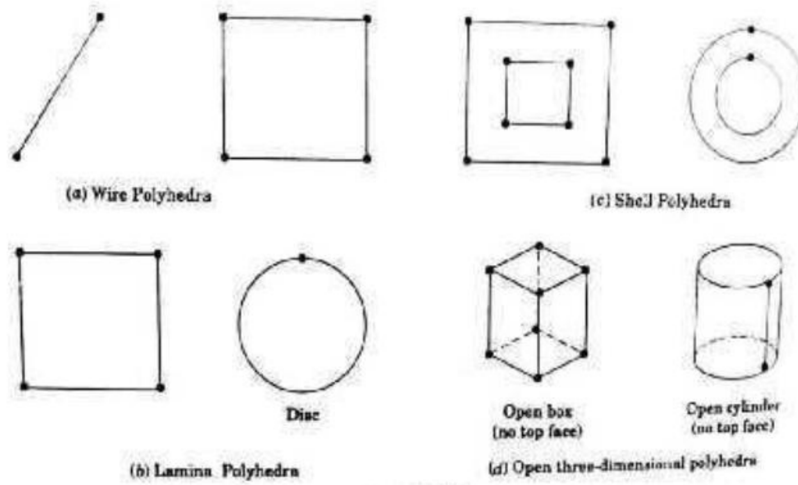
- A solid is represented as a collection of connected surface elements, the boundary between solid and non-solid.

- Boundary representation models are composed of two parts:

  - Topology, and
  - Geometry (surfaces, curves and points).

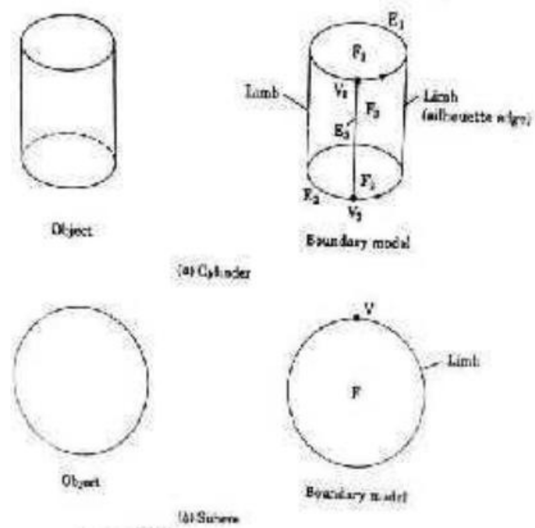
- A minimum body is a point; topologically this body has one face, one vertex, and no edges. It is called a seminal or singular body

Open polyhedral objects



Open polyhedral objects.

Curved Objects



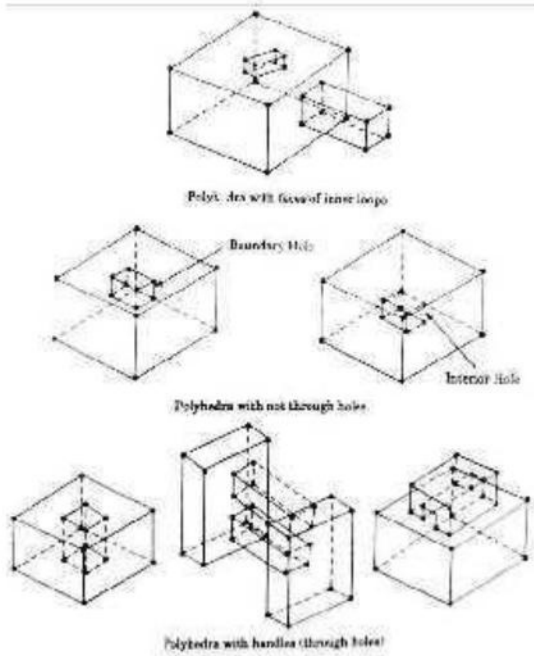
Exact B-rep of a cylinder and a sphere.

- **Euler's formula**

  Euler – Poincare Law for closed objects : $F - E + V - L = 2\,(B - G)$

  Euler – Poincare Law for open objects : $F - E + V - L = B - G$

Solid Model Generation using B-rep



Advantages of b-rep

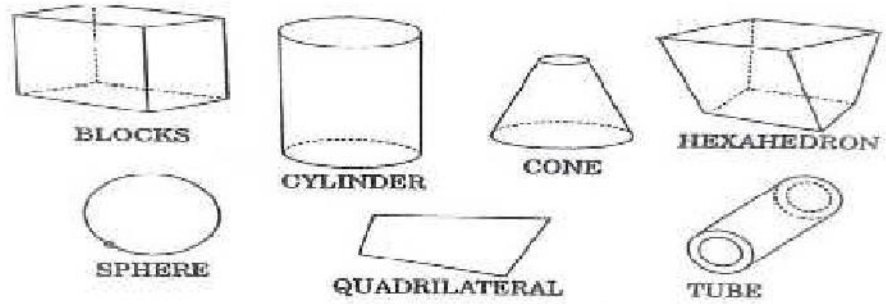
- O Appropriate to construct solid models of unusual shapes
- O Relatively simple to convert a b-rep model to wireframe model

Disadvantages of b-rep

- O Requires more storage
- O Not suitable for applications like tool path generation O Slow manipulation

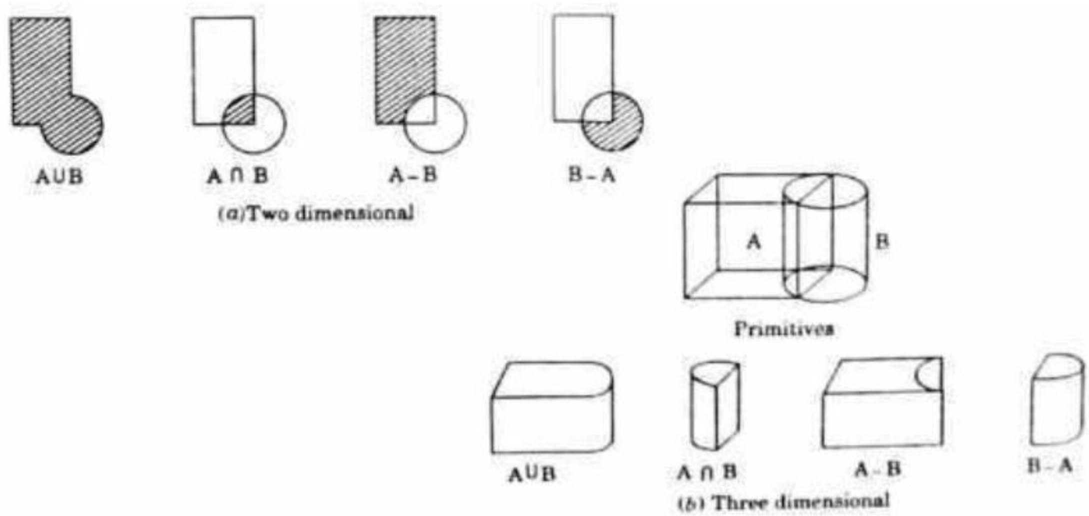
CONSTRUCTIVE SOLID GEOMETRY (CSG and C-rep)

- Constructive solid geometry (CSG) (formerly called computational binary solid geometry) is a technique used in solid modeling.
- Constructive solid geometry allows a modeler to create a complex surface or object by using Boolean operators to combine objects.
- Often CSG presents a model or surface that appears visually complex, but is actually little more than cleverly combined or de-combined objects
- The simplest solid objects used for the representation are called **primitives**. Typically they are the objects of simple shape:
  - O cuboids
  - O cylinders
  - O prisms O pyramids
  - O spheres
  - O cones

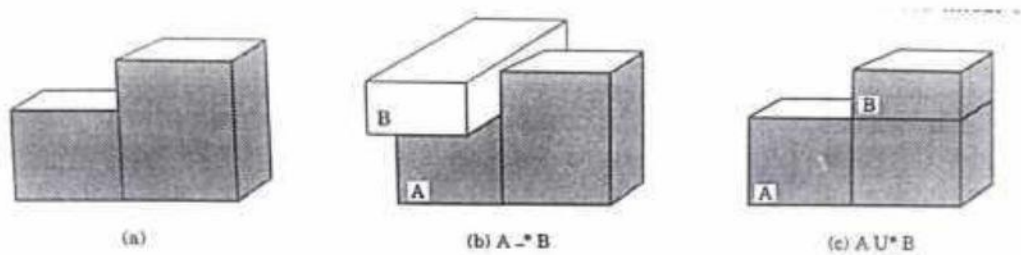


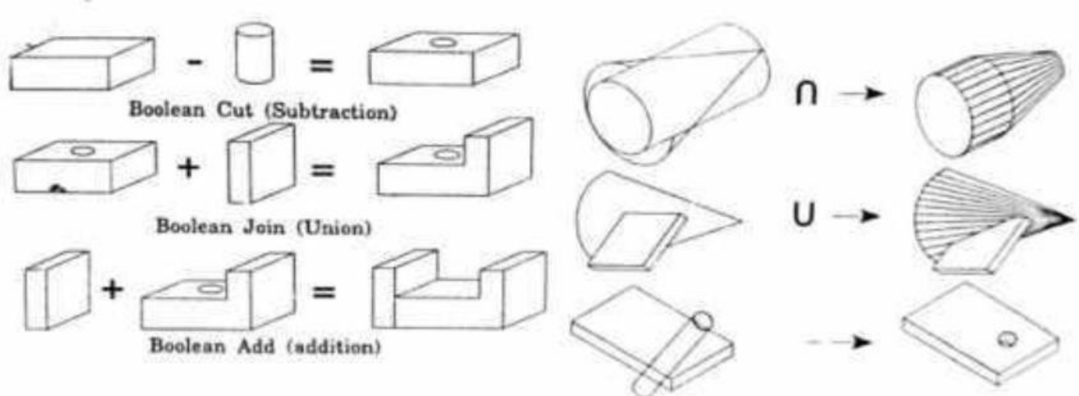
Solid primitives

Boolean Operations

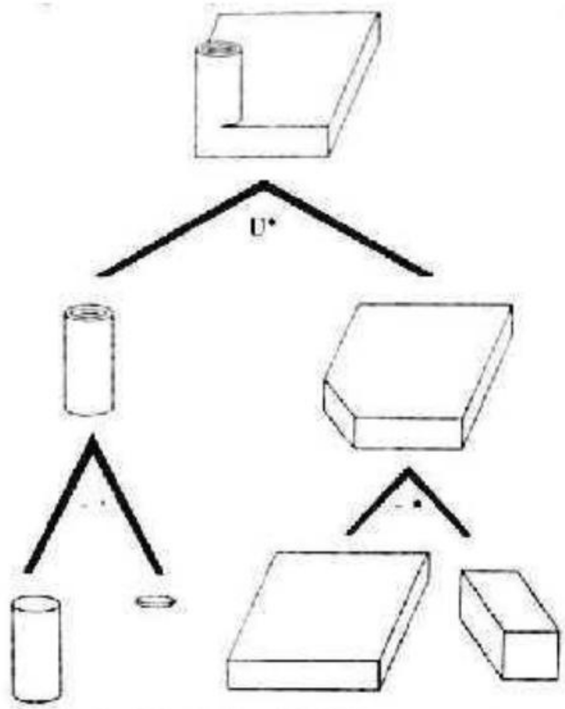


Boolean operations of a block *A* and cylinder *B*.





CSG Tree



An object defined by CSG and its tree.