IPE-409 CAD/CAM

Chapter-3

Techniques for Geometric Modelling

Representation of curves

- Mathematically straightforward geometric entities are curves and their representations are most complete.
- Surfaces are extension of curves.

$y=mx + c$ (1)

$ax+by+c=0$ (2)

$ax^2+by^2+2kxy+2fx+2gy+d=0$ (3)

Why are these equations are not adequate for CAD?

- Unbounded geometry
- Multi-valued
- Sequence of points not available
- Equation changes with coordinate system
- Difficulties in faired shapes representation, intersections between solid or surfaces

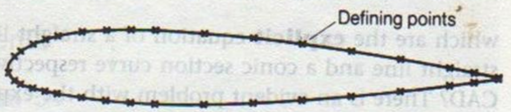


FIGURE 3.1
An aerofoil section.

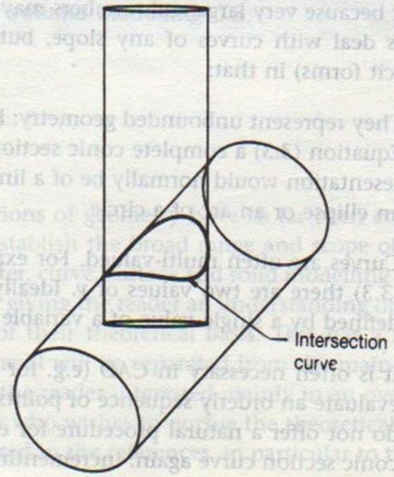


FIGURE 3.2
Intersection of two cylinders.

Parametric Representation of geometry

- The parametric representation of geometry essentially involves expressing relationships for the x, y and z coordinates of points on a curve or surface or a solid not in terms of each other but of one or more independent variables known as parameters.

  - For curve a single parameter is used: x, y and z are express in terms of a single variable typically u
  - For surface two parameters u and v
  - For solid three parameters u, v and w

Parametric Representation of geometry

- Position of any point on a space curve can be expressed as

$$p = p(u)$$

which is same as x=x(u), y= (u), and z=z(u)

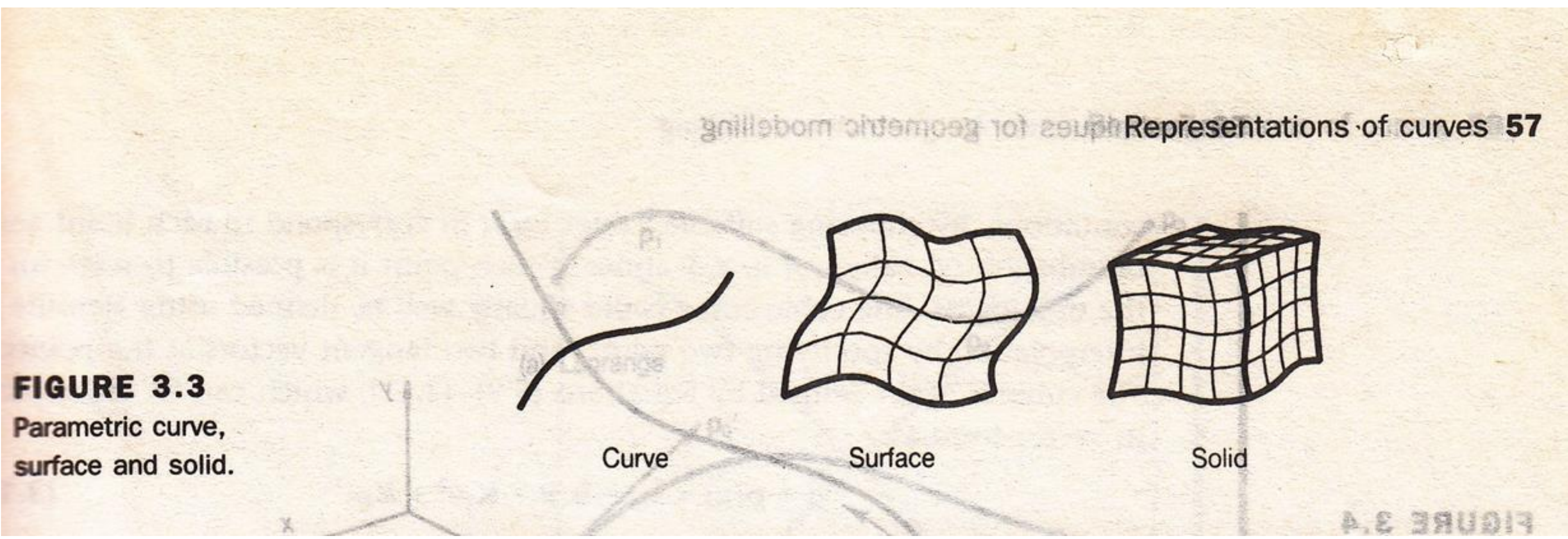


FIGURE 3.3
Parametric curve, surface and solid.

Parametric cubic polynomial curves

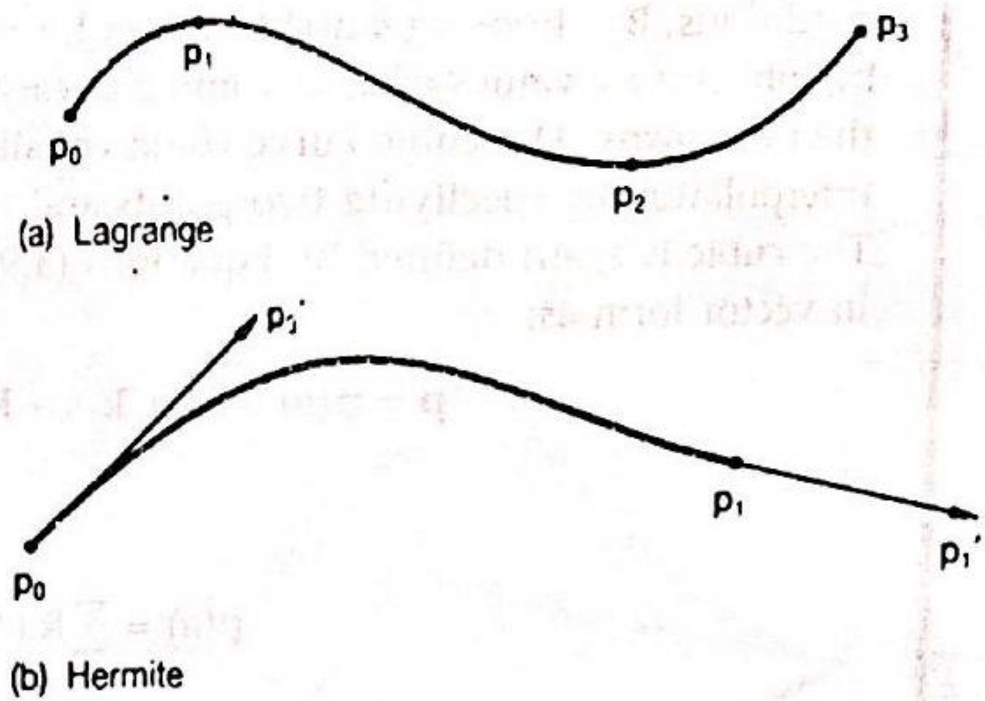


FIGURE 3.5
Lagrange and Hermite interpolation.

Parametric cubic polynomial curves

- **Vector form of cubic curve:**

$$\mathbf{p} = \mathbf{p}(u) = \mathbf{k}_0 + \mathbf{k}_1 u + \mathbf{k}_2 u^2 + \mathbf{k}_3 u^3 \qquad (3.12)$$

$$\mathbf{p}(u) = \sum_{u=0}^{3} \mathbf{k}_i u^i$$

$$\mathbf{p}' = \mathbf{p}'(u) = \mathbf{k}_1 + 2\mathbf{k}_2 u + 3\mathbf{k}_3 u^2 \qquad (3.13)$$

Parametric cubic polynomial curves

- **Vector form of cubic curve:**

Using the end points $\mathbf{p}_0$ and $\mathbf{p}_1$, and the end slopes $\mathbf{p}'_0$ and $\mathbf{p}'_1$, we can substitute in Equations (3.12) and (3.13) to derive the unknowns. It is usual to assign $u = 0$ and $u = 1$ to the two ends of the segment, with $0 < u < 1$ between. Thus:

$$
\begin{aligned}
\mathbf{k}_0 &= \mathbf{p}_0 \\
\mathbf{k}_0 + \mathbf{k}_1 + \mathbf{k}_2 + \mathbf{k}_3 &= \mathbf{p}_1 \\
\mathbf{k}_1 &= \mathbf{p}'_0 \\
\mathbf{k}_1 + 2\mathbf{k}_2 + 3\mathbf{k}_3 &= \mathbf{p}'_1
\end{aligned}
\qquad (3.14)
$$

Parametric cubic polynomial curves

- **Vector form of cubic curve:**

Solving for $\mathbf{k}_0$ to $\mathbf{k}_3$ we obtain:

$$\begin{aligned}
\mathbf{k}_0 &= \mathbf{p}_0 \\
\mathbf{k}_1 &= \mathbf{p}_0' \\
\mathbf{k}_2 &= 3(\mathbf{p}_1 - \mathbf{p}_0) - 2\mathbf{p}_0' - \mathbf{p}_1' \\
\mathbf{k}_3 &= 2(\mathbf{p}_0 - \mathbf{p}_1) + \mathbf{p}_0' + \mathbf{p}_1'
\end{aligned} \tag{3.15}$$

Thus, by substitution in Equation (3.12), we obtain:

$$\begin{aligned}
\mathbf{p} = \mathbf{p}(u) = {} & \mathbf{p}_0(1 - 3u^2 + 2u^3) + \mathbf{p}_1(3u^2 - 2u^3) + \\
& \mathbf{p}_0'(u - 2u^2 + u^3) + \mathbf{p}_1'(-u^2 + u^3)
\end{aligned} \tag{3.16}$$

Parametric cubic polynomial curves

- **Hermit cubic curve**

Equation (3.16) from the boxed section above gives the general form of a cubic polynomial in the Hermite basis as:

$$\mathbf{p} = \mathbf{p}(u) = \mathbf{p}_0(1 - 3u^2 + 2u^3) + \mathbf{p}_1(3u^2 - 2u^3) + \mathbf{p}_0'(u - 2u^2 + u^3) + \mathbf{p}_1'(-u^2 + u^3) \quad \mathbf{(3.16)}$$

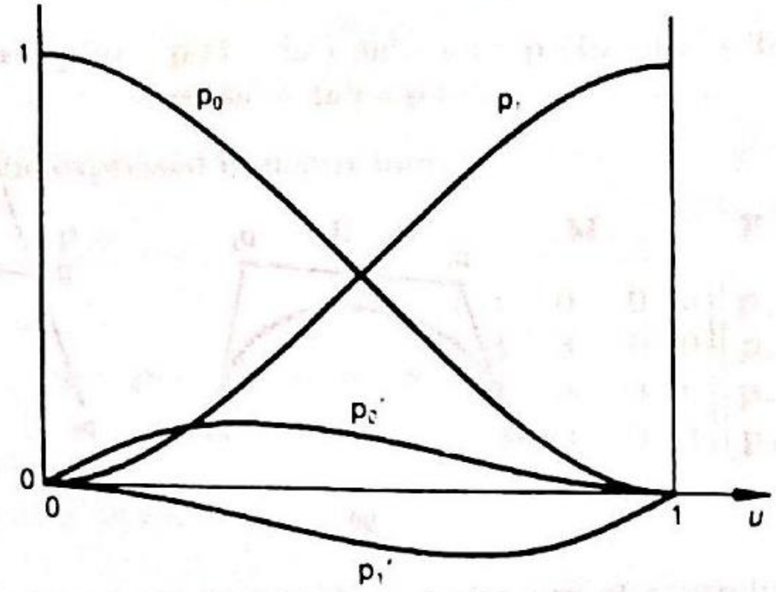


FIGURE 3.6
Blending functions for a Hermite cubic curve.

Parametric cubic polynomial curves

Calculation of a Hermite curve

As an example of the use of Hermite interpolation, let us calculate the parametric mid-point of the Hermite cubic curve that fits the points $\mathbf{p}_0 = (1, 1)$, $\mathbf{p}_1 = (6, 5)$ and the tangent vectors $\mathbf{p}'_0 = (0, 4)$, $\mathbf{p}'_1 = (4, 0)$. At the parametric mid-point, $u = 0.5$ (recall that u varies in the range 0 to 1 along the curve). Substituting this value and the values for $\mathbf{p}_0$, $\mathbf{p}_1$, $\mathbf{p}'_0$, and $\mathbf{p}'_1$ into Equation (3.16), we obtain:

$$\begin{aligned} x(0.5) &= 1[1 - 3(0.5^2) + 2(0.5^3)] + 6[3(0.5^2)] - 2(0.5^3)] + \\ &\quad 0[0.5 - 2(0.5^2) + (0.5^3)] + 4[-(0.5^2) + (0.5^3)] \\ &= 1 \times 0.5 + 6 \times 0.5 + 0 \times 0.125 - 4 \times 0.125 \end{aligned}$$

and:

$$\begin{aligned} y(0.5) &= 1[1 - 3(0.5^2) + 2(0.5^3)] + 5[3(0.5^2) - 2(0.5^3)] + \\ &\quad 4[0.5 - 2(0.5^2) + (0.5^3)] + 0[-(0.5^2) + (0.5^3)] \\ &= 1 \times 0.5 + 5 \times 0.5 + 4 \times 0.125 - 0 \times 0.125 \end{aligned}$$

which give $\mathbf{p}(0.5) = (3, 3.5)$.

Bezier Curves

Bézier used a **control polygon** for curves, in place of points and tangent vectors (Figure 3.7(a)). This polygon is **approximated** by a polynomial curve whose degree is one less than the number of polygon vertices (which are also known as **control points** or track points). Figure 3.7(a) shows a four-point polygon which is approximated by a cubic curve in which $\mathbf{p}_0$ and $\mathbf{p}_3$ are equivalent to $\mathbf{p}_0$ and $\mathbf{p}_1$ for the Hermite basis cubic polynomial. The mid-vertices $\mathbf{p}_1$ and $\mathbf{p}_2$ are defined to be 1/3 of the way along the tangent vectors at $\mathbf{p}_0$ and $\mathbf{p}_3$ respectively (this is shown in Figure 3.7(b);

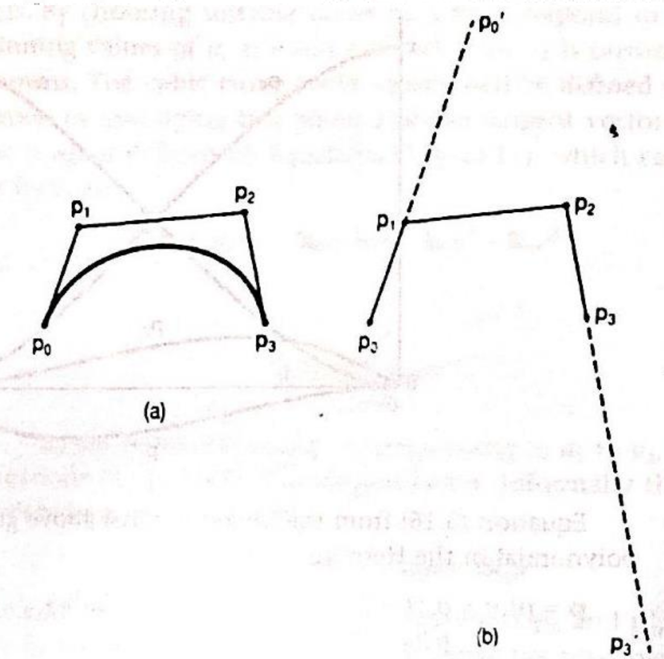


FIGURE 3.7
Cubic Bézier curves.

Bezier Curves

Bézier polynomial curves

For the Bézier cubic polynomial, referring to Figure 3.7(b), we can write:

$$\mathbf{p}_0' = 3(\mathbf{p}_1 - \mathbf{p}_0) \quad (3.18)$$

$$\mathbf{p}_3' = 3(\mathbf{p}_3 - \mathbf{p}_2) \quad (3.19)$$

$$\mathbf{p} = \mathbf{p}(u) = \mathbf{p}_o(1-3u+3u^2-u^3)+\mathbf{p}_1(3u-6u^2+3u^3)+\mathbf{p}_2(3u^2-3u^3)+\mathbf{p}_3(u^3)$$

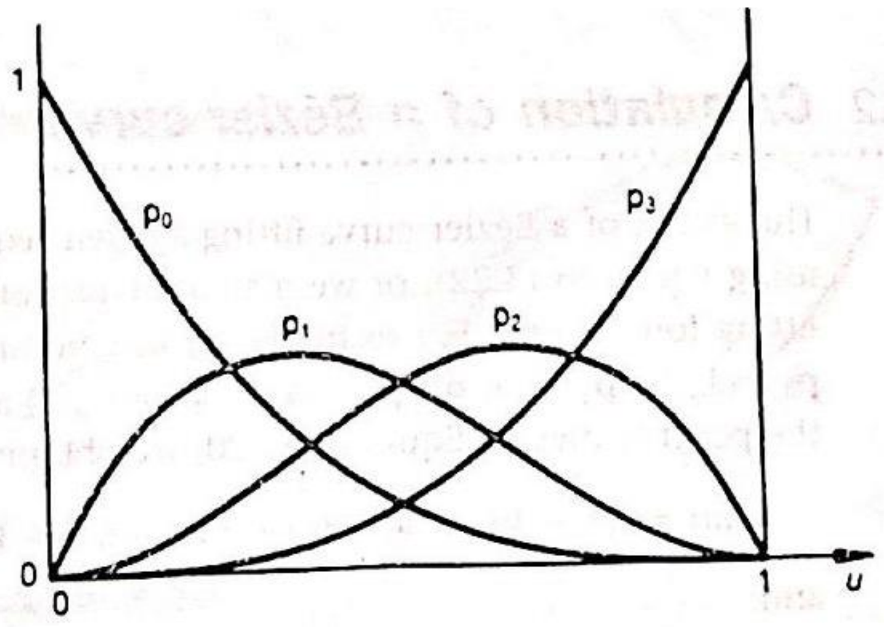


FIGURE 3.8
Blending functions for a cubic Bézier curve.

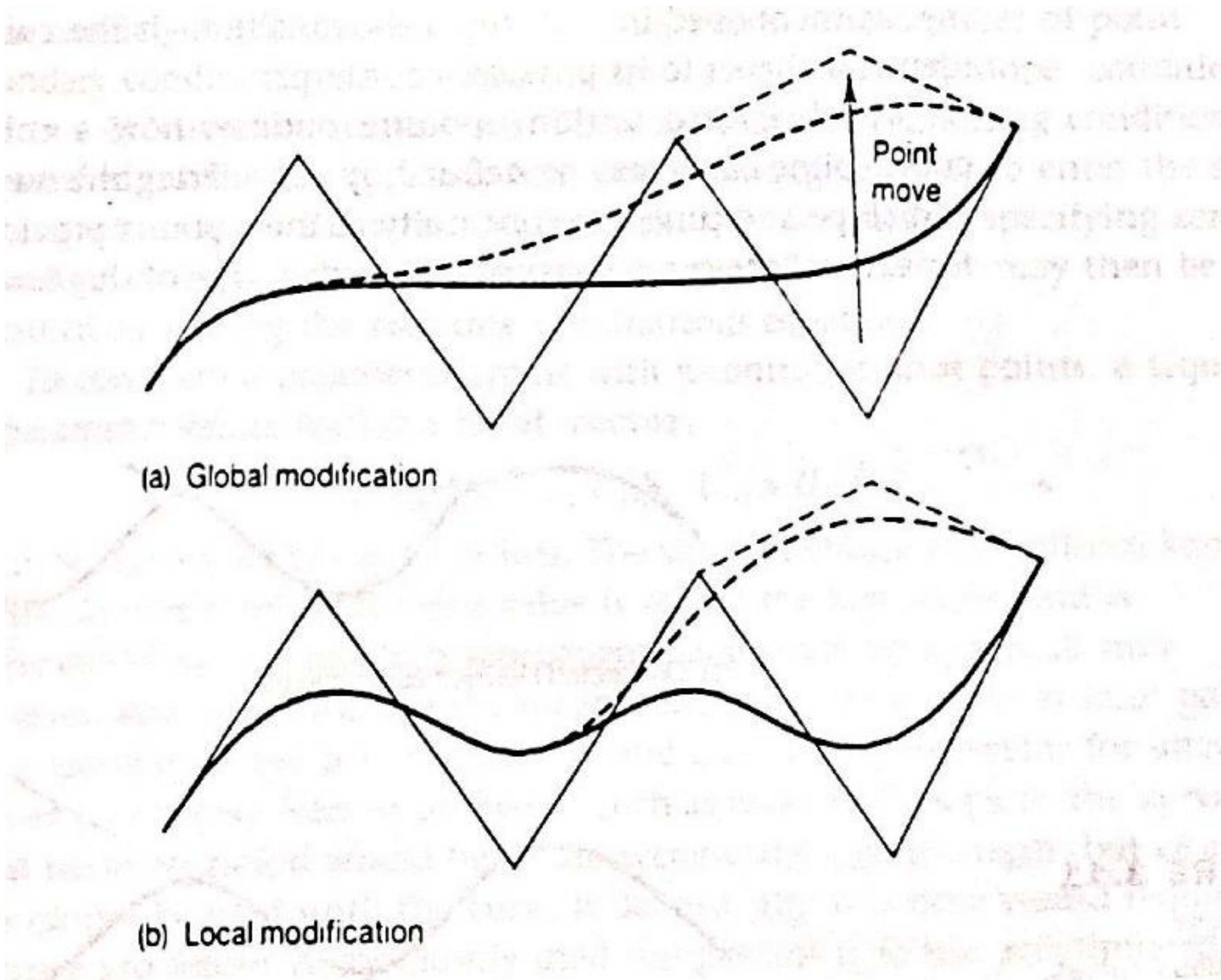
Bezier Curves

Calculation of a Bézier curve

The values of a Bézier curve fitting a given sequence of points may be calculated by using Equation (3.22), or we may use Equation (3.20) directly if the curve is cubic – fitting four points. For example, let us compute the values of a curve fitting points $\mathbf{p}_0 = (1, 1)$, $\mathbf{p}_1 = (3, 6)$, $\mathbf{p}_2 = (5, 7)$, $\mathbf{p}_3 = (7, 2)$ at $u = 0.4$ and $u = 0.6$. Substituting for the point values in Equation (3.20) we obtain:

p(0.4) = (3.4, 4.952)
p(0.6) = (4.6, 5.248)

Local modification & Global modification



(a) Global modification

(b) Local modification

B-Spline Curves

Neither the Bézier nor the cubic spline curve formulations allow local modification of curves, and Bézier polynomials are, in addition, somewhat constrained in the number of points that they may approximate without the degree of the curve becoming inconveniently high. Both of these limitations are overcome by a generalization of the Bézier approach known as the **B-spline** method, which again uses blending functions to combine the influence of a series of control or track points in an approximate curve. For a series of $n + 1$ points $\mathbf{p}_i$, the formulation is:

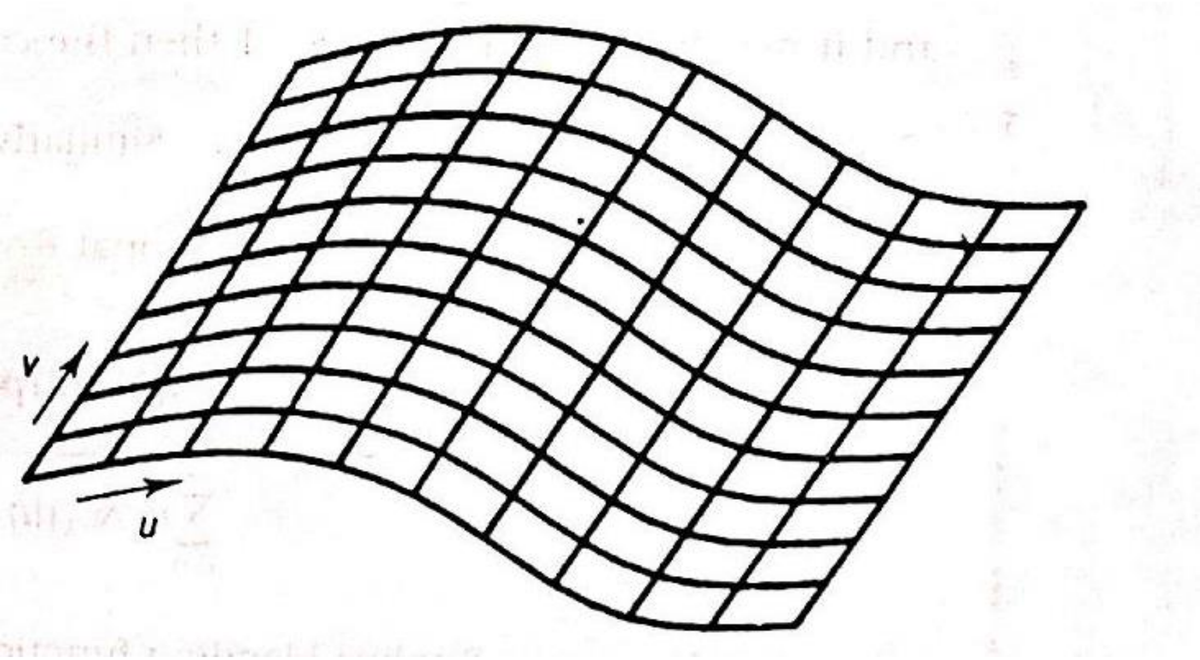
$$\mathbf{p}(u) = \sum_{i=0}^{n} N_{i,k}(u)\mathbf{p}_i \qquad (3.24)$$

where the B-spline blending functions are $N_{i,k}$.

Techniques for Surface Modelling

Surface patch

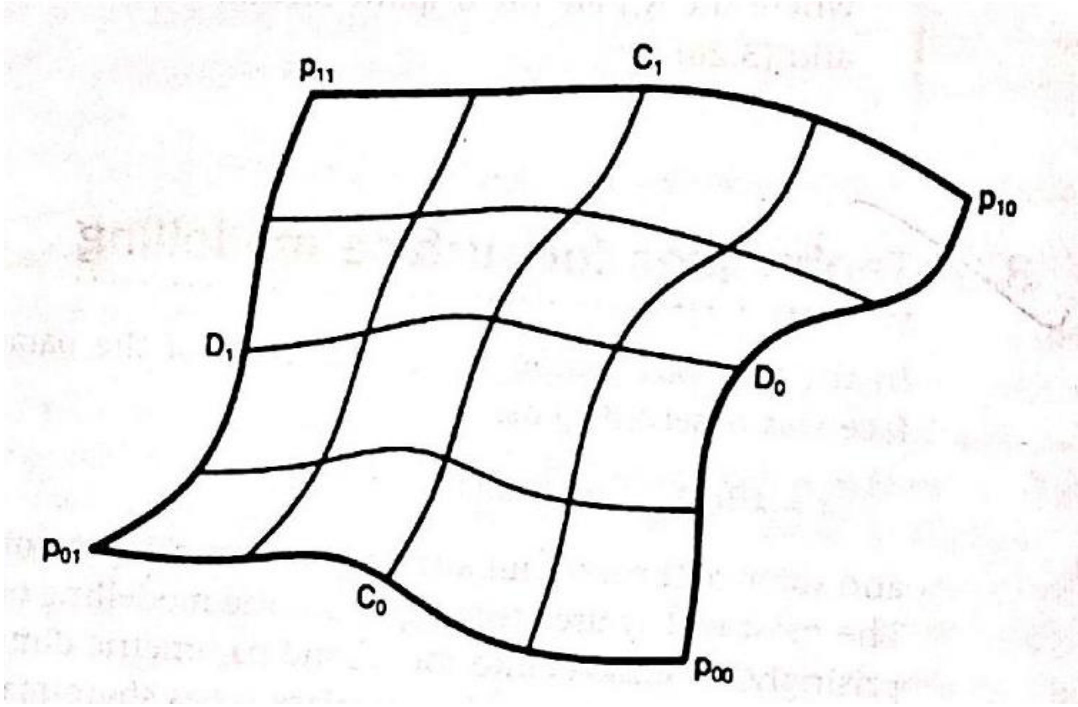
- Biparametric
- Isoparametric



Techniques for Surface Modelling

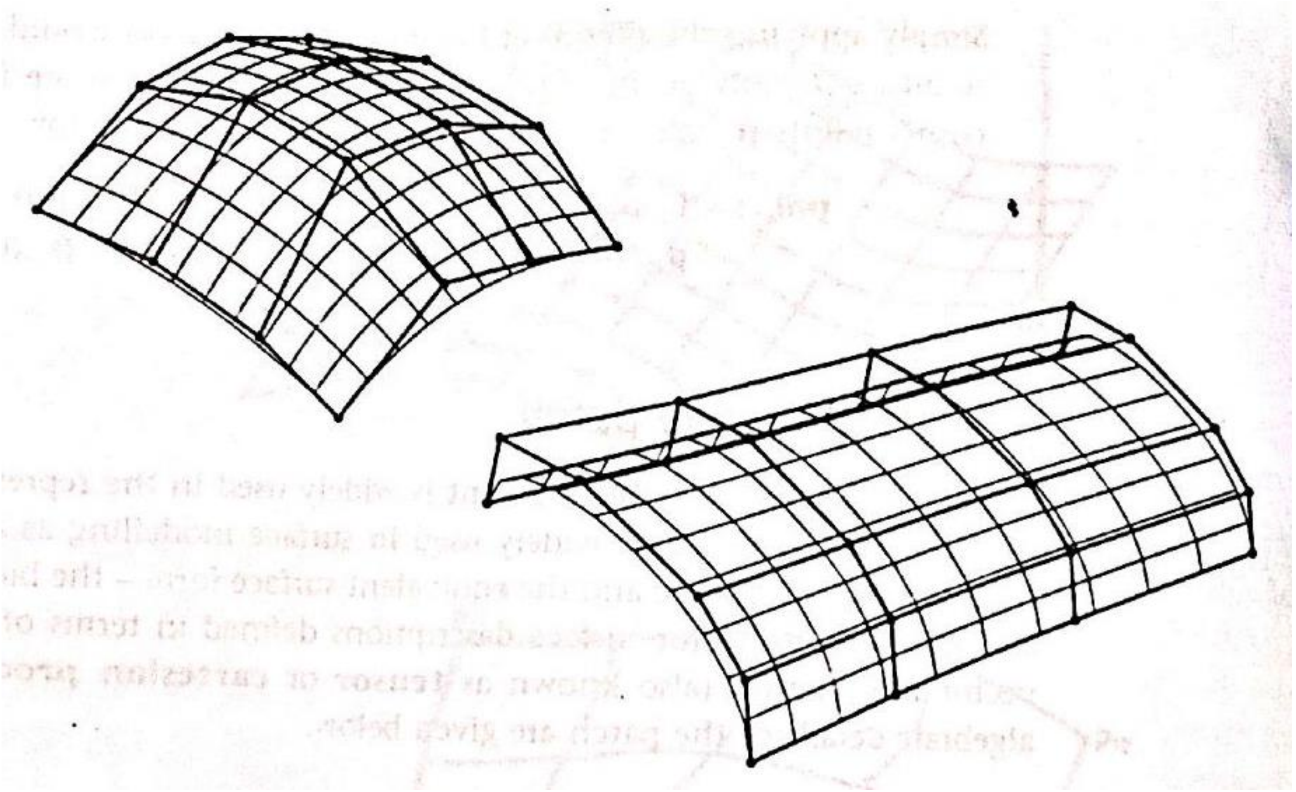
Coons patch

- Sculptured surface



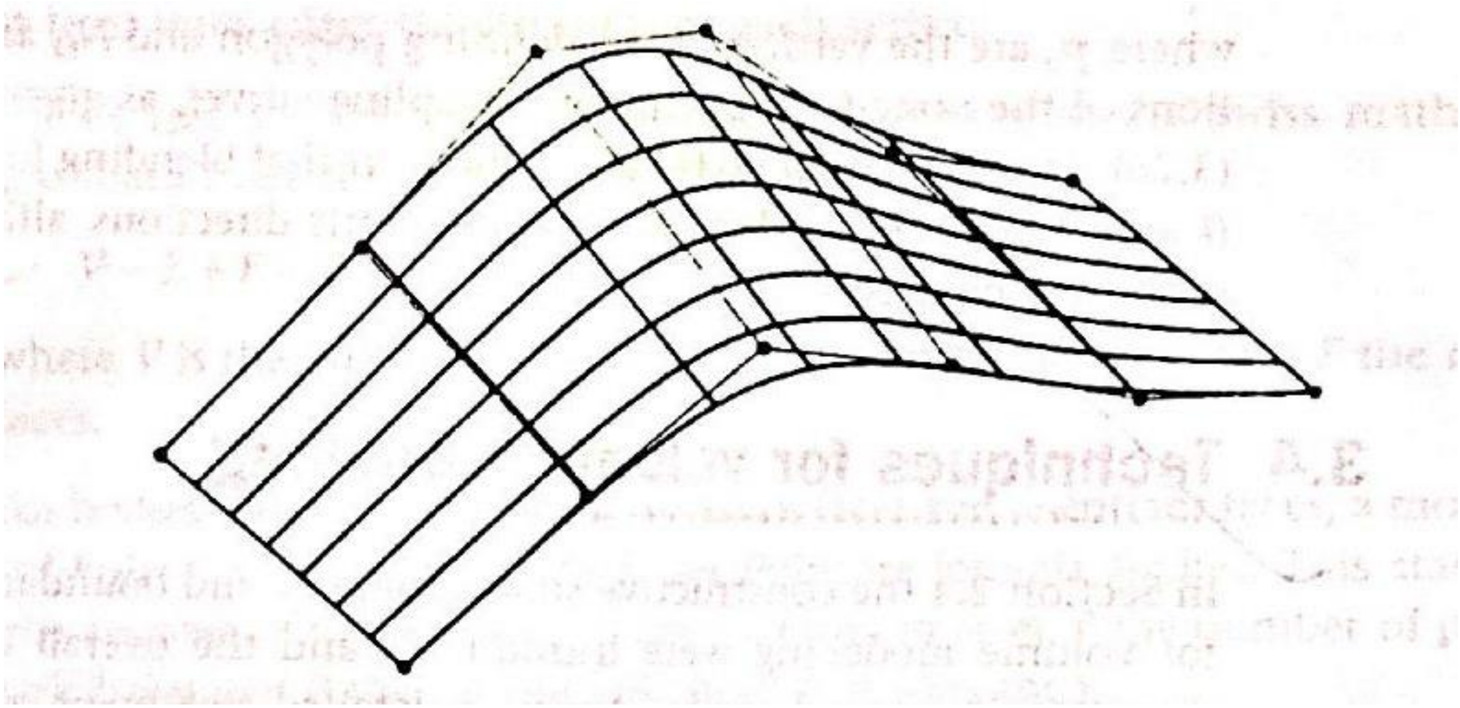
Techniques for Surface Modelling

Bezier surface



Techniques for Surface Modelling

B-Spline surface



Techniques for Volume Modelling

Boundary models

Euler's rule

$$V - E + F = 2$$

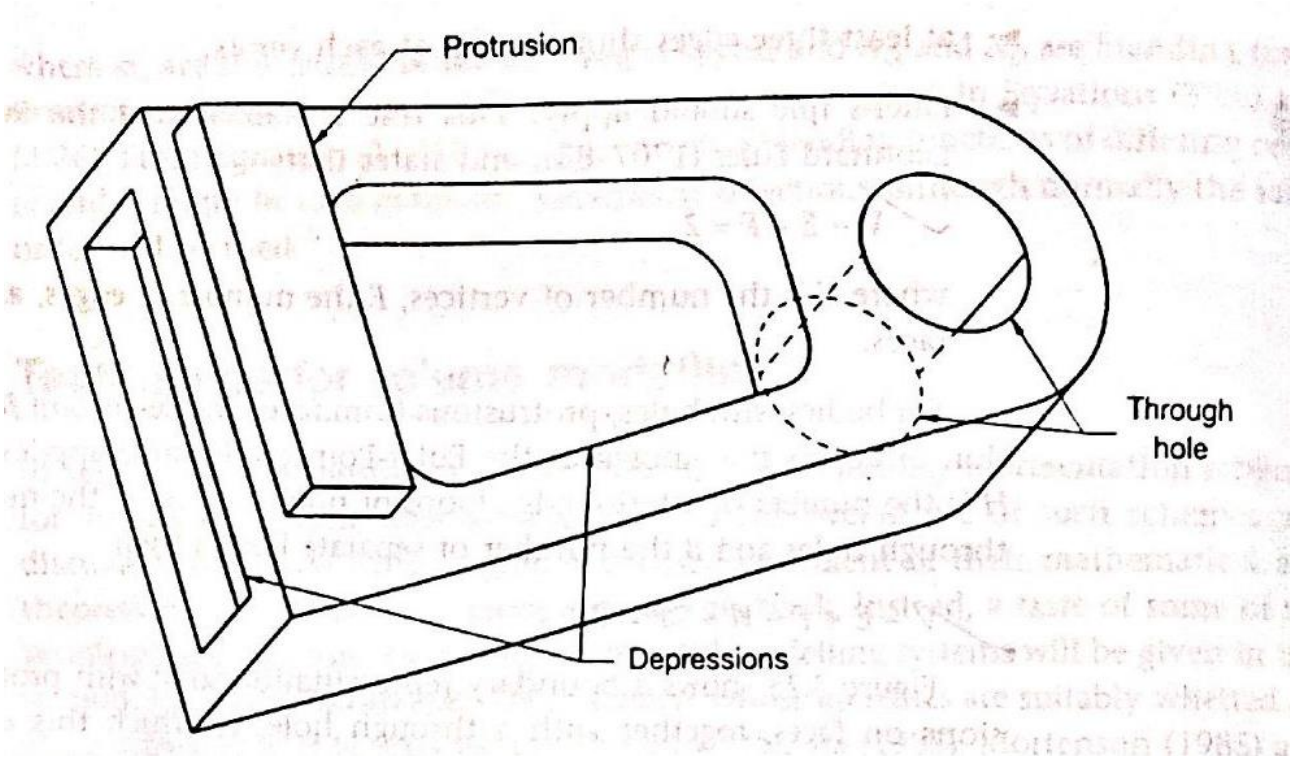
where V is the number of vertices, E the number of edges, and F the number of faces.

For bodies with holes, protrusions from faces and re-entrant faces, a modified version of Euler's rule known as the Euler-Poincaré formula applies. This states that, if H is the number of interior edge loops or holes in faces, P the number of passages or through holes and B the number of separate bodies then:

$$V - E + F - H + 2P = 2B$$

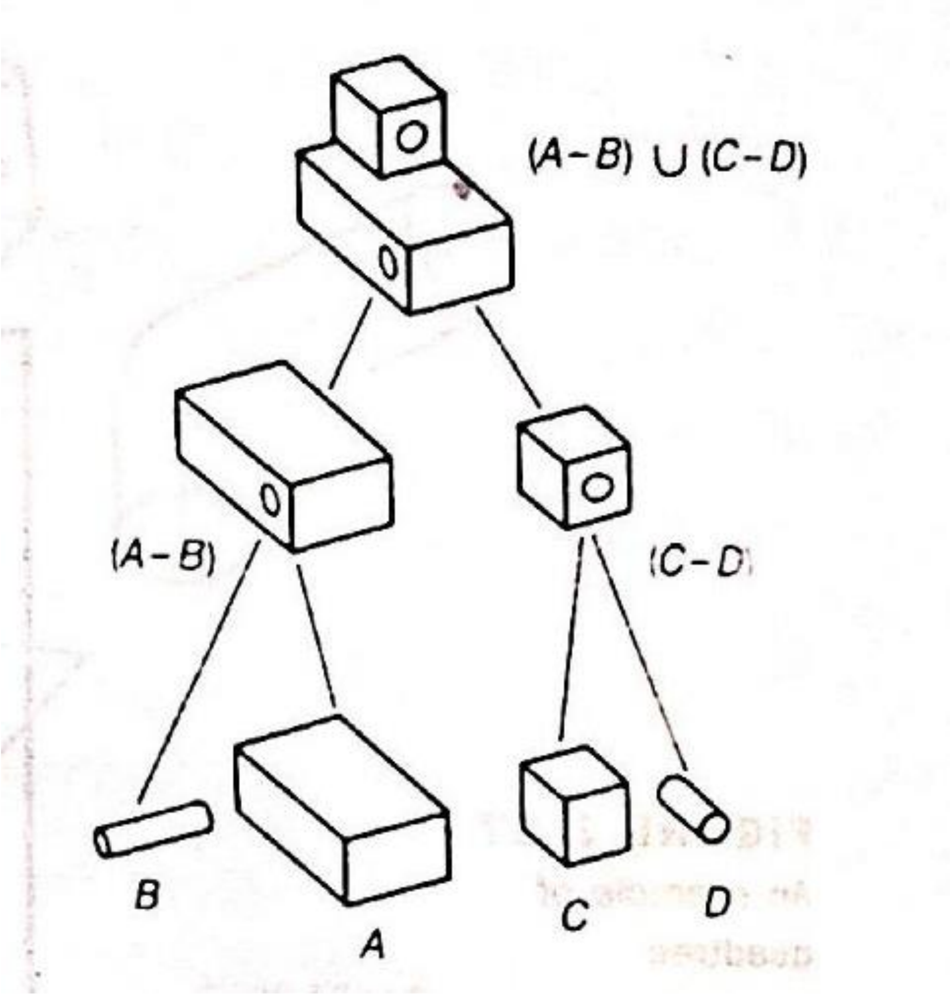
Techniques for Volume Modelling

Boundary models



Techniques for Volume Modelling

CSG



Techniques for Volume Modelling

Other modelling techniques

- Pure primitive instancing
- Cell decomposition
- Spatial occupancy enumeration

Reference book

- CAD CAM principles, practice and manufacturing management *by Chris McMohan and Jimmie Browne*

Thank You