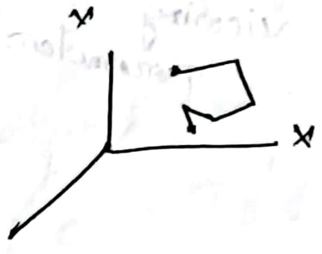
Geomatric Representation

Planar polygon ?

→ It is a polygon in which all vertices lie on the same plane



Polyhedron :

It is a 3D shape made of flat polygonal faces that enclose a volume.

In modeling polyhedrons are often treated as blocks rather than wire frames

Curved surface design

It is used in 3D modeling to create more realistic shapes.

modeling Approaches

- use small ~~ps~~ curved patches joined together
- Bulid model with solid shapes

model construction method

Additive	Subtractive
simple shapes to create	remove parts from an object

Wire-frame models (4)

It is a model consist of edges, vertices and polygon here vertices are connected by edge and polygons are sequences of vertices or edges.

So, its also called polygonal net / mesh.

Advantage	Disadvantage
simple and easy to create	Lack of detail and realism
provide clear view of the object structure	complex model can become complex
requires minimal storage space	unsuitable for simulations.

Designer needs little training	Image causes confusion
System needs little memory	Can't get required information from this model.
Efficient for visualization	Limited representation surface and materials.

Representing polygonal Net model

Different way representing polygonal net model

Some of are →

Vertex list representation: Simplest way. It stores a list of vertices each of which has a position in 3D space.

Explicit vertex list (SAT): Vertices V (P0, P1, P2 ... Pn) are stored in the order they appear

Shared vertices are repeated and edges may be drawn multiple times

Polygon Listing: Each vertex is stored once in a list $V(P_0 - - P_N)$ and polygons reference these vertices using indices.

here also shared edges drawn multiple times

Explicit Edge Listing

vertex list - Each vertex stored once.

Edge " - each edge is stored once linking 2 vertices

polygon - Defined as indices pointing to edges

Benifit - Reduces redundancy and each edge draw only once.

This method is efficient for Wireframe model

Syllabus mein not important

problem of interpolation & Approximation

Interpolation problem

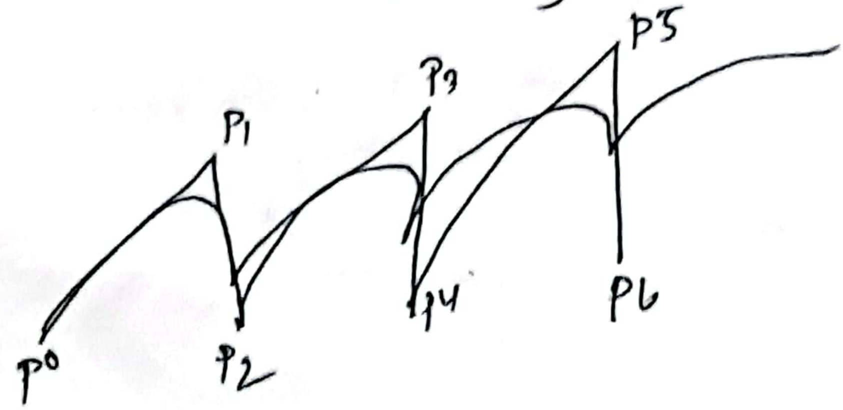
It involves creating a curve or surface that pass through given set of data points.

Challenges are overfitting, irregular data and high computation — large dataset a expensive.

Approximation Problem

Definition: It creates a curve on surface that closely follows the data points without necessarily passing through them

Challenges are ⇒ Loss of precision, Balancing accuracy, Bais



Seg 7 (chp 10)

Hidden Surface

Hidden surface is a parts of object that are not visiable from a particuler viewpoint because they are blocked by other objects in the scene.

* In overlap 2 point lie on the same projection line they are said to overlap in the projection

* If 2 point overlap a depth comparison is perfon need to determine which point is in front of the other

depth comparison → In the same projection line

$\underline{x_1 = x_2}$ and $\underline{y_1 = y_2}$ হলে depth comparison হবে

Hiddien surface algorithm used for which objects and surface will obscure (dark) determine for

Z buffer Algorithm

The z buffer Algorithm is used for hidden surface removal in 3D graphics. It works by sorting the depth of each pixel to determine which surface are visible.

When the surface is drawn, its depth at each pixel is compared with the depth value already sorted in the z-buffer for pixel

- If the new depth smaller (close to camera) then the sorted depth, the surface visible

- If the new depth is larger, the surface hidden and the z buffer ~~stays~~ remain unchanged.

Z buffer ensures that only the closest surfaces are visible in each pixel.

Steps in Z buffer Algorithm.

• Initialize the Z buffer.

① Set the Screen background Colour (black)

② Initialize the Z buffer to the minimum depth for each pixel.

Process Each object

• Each object calculate its depth each pixel.

• If the new depth smaller, update Z buffer changes pixel clr

• If the new depth Larger, keep pixel unchanged.

Final Image.

Display the pixels in the frame buffer, using depth values in the Z buffer to determine which surfaces are visible, hidden surface are ramoved.

Example (Z buffer)

Consider Looking at a 3D scene with a

~~Cube and sphere~~.

tree, house, Car in 7, 5, 3, depth.

Steps

- Initialize background clr black.
- " Z buffer minimum depth (10)

- Compare depth value 10, in tree(7) smaller than 10.

 So update clr pixel is tree clr

- Compare update (7 tree) depth value. here (5 house) smaller than (7 tree).

 So update clr pixel is house clr.

Same car (3) less than (5 house)

Final Image

Car front → house → tree

LR

Subdivision or Warnock Algo

It is a recursive method that subdivides the screen into smaller quadrants until each quadrant can be easily classified as either visible or hidden

Steps

- Set the viewing area to the whole screen
- In PVL sort polygon by depth (z Value)

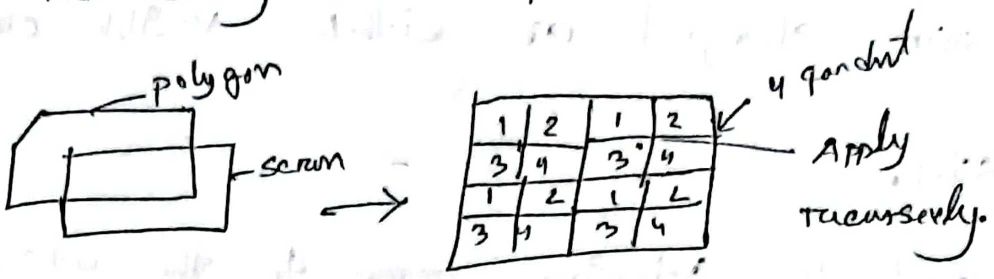
- Classify polygons
  - Complete inside the viewing area.
  - " outside.
  - Surrounding the area.
  - Partially overlap

- Remove polygons
  - remove outside polygons
  - " polygons surrounded by others.
  - " disjoint polygons.

Check PVL → If empty area hidden.
If one polygons remain — It visible

• If multiple polygons remain, Subdivide 4 Quadrants.

• Apply recursively each Quardent.



— x —

Math 10.1 (which point obscure)

Given points.

$C\ (0,\ 0,\ -10)$
$\quad x_0\ \ y_0\ \ z_0$

$P_1\ (1,\ 2\ 0)$
$\quad x_1\ \ y_1\ \ z_1$

$P_2\ (3,\ 6,\ 20)$
$\quad x_2\ \ y_2\ \ z_2$

$P_3\ (2,\ 4,\ 6)$
$\quad x_3\ \ y_3\ \ z_3$

Solⁿ $P_0(x_0, y_0, z_0)$ & $P_1(x_1, y_1, z_1)$

$x = x_0 + (x_1 - x_0)\, t$

$y = y_0 + (y_1 - y_0)\, t$

$z = z_0 + (z_1 - z_0)\, t$

now eq for $C(0, 0, -10)$ and $P_1(1, 2, 0)$
$\quad\; x_0 \;\; y_0 \;\; z_0 \qquad\qquad\qquad x_1 \;\; y_1 \;\; z_1$

$$x = x_0 + (x_1 - x_0)t$$
$$= 0 + (1 - 0)t = t$$

$$y = y_0 + (y_1 - y_0)t$$
$$= 0 + (2 - 0)t = 2t$$

$$z = -10 + (0 - -10)t$$
$$= -10 + 10t$$

So

$x = t$
$y = 2t$
$z = -10 + 10t$

put values on P2 (3) $\Rightarrow$ $x = 3$, $y = 6$, $z_2 = 20$

put P3 (2) $\Rightarrow$ $x = 2$, $y = 4$, $z = 10$

put valu P1 (1)

$x = 1$
$y = 2$
$z_1 = 0$

(এখানে P3 doesn't obscure
কারণ $-z = 10$ থাকে P3 z ⑥ Given)

here P1, P2 takes P3 বাদ

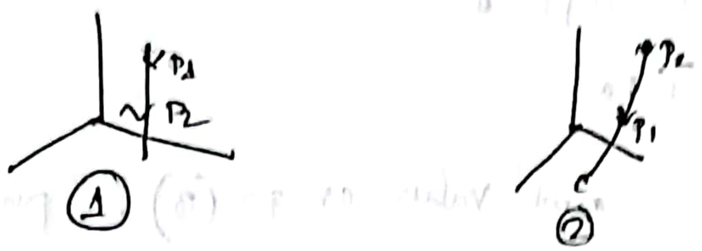
$z_1 < z_2$ So P1 infront of P2.
$0 < 20$

P1 obscure of P2

10.5

Steps of weather any given point obscure other point :

① weather 2 point lie same projection line
② If they do, which point in font of the other



10.10

Basic Scan Line Method for hidden surface

Scan line method process the screen row by row to determine which surface are visible or hidden.

algorithm looks all polygon intersect (y=a) line
for each pixel one the line it compares z value for each polygon
with smallest z value set the color