Writing a ray tracer

What coordinate system?

- **<u>world space</u>, not eye space**
  - viewing transformation would only simplify 1st ray
  - objects change shape in perspective, so intersections are harder
  - perspective is not angle-preserving, so bounces are harder

Ray casting algorithm

```
for each pixel (or sample)
    for each primitive
        test for ray-primitive intersection
    select frontmost intersection
    trace new rays (recursively) in reflection and refraction directions
    compute color
```

Two ways to define a geometric primitive

Implicit form

$$F(x,y,z)\begin{cases} <0 & inside \\ =0 & on \\ >0 & outside \end{cases}$$

Parametric form

$$(\, x(u,v),\, y(u,v),\, z(u,v)\,)$$

Example: a plane

implicit

$$ax + by + cz + d = 0$$

$$\mathbf{N} \bullet \mathbf{P} = 0$$

parametric

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} = u \begin{bmatrix} x_u \\ y_u \\ z_u \end{bmatrix} + v \begin{bmatrix} x_v \\ y_v \\ z_v \end{bmatrix} + \begin{bmatrix} x_0 \\ y_0 \\ z_0 \end{bmatrix}$$

Ray-surface intersection

$$f(x,y,z)=0 \quad \xrightarrow{\begin{array}{l} x = x_0 + t\Delta x \\ y = y_0 + t\Delta y \\ z = z_0 + t\Delta z \end{array}} \quad f^*(t)=0$$

1. **Substitute parametric form of ray equation into implicit form of surface equation**

2. **Find *positive*, *real* roots**

Univariate root finding

- **closed form solution**
- **Newton's method**
- ***regula falsa***
- **interval methods**

(from Hanrahan)

Ray-sphere intersection

Sphere $(x-a)^2+(y-b)^2+(z-c)^2-r^2=0$

Ray

$$x = x_0 + t\Delta x$$

$$y = y_0 + t\Delta y$$

$$z = z_0 + t\Delta z$$

Substituting gives $At^2+Bt+C=0$

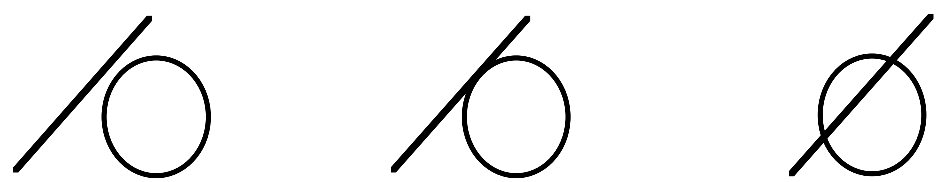
where

$$A = \Delta x^2 + \Delta y^2 + \Delta z^2$$

$$B = 2(\Delta x(x_0-a)+\Delta y(y_0-b)+\Delta z(z_0-c))$$

$$C = (x_0-a)^2+(y_0-b)^2+(z_0-c)^2-r^2$$

Solution $t = \frac{-B \pm \sqrt{B^2 - 4AC}}{2A}$



If discriminant **<0** **=0** **>0**

Point of intersection

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} x_0 \\ y_0 \\ z_0 \end{bmatrix} + t \begin{bmatrix} \Delta x \\ \Delta y \\ \Delta z \end{bmatrix}$$

Ray-plane intersection

Plane $ax + by + cz + d = 0$

Ray

$$x = x_0 + t\Delta x$$
$$y = y_0 + t\Delta y$$
$$z = z_0 + t\Delta z$$

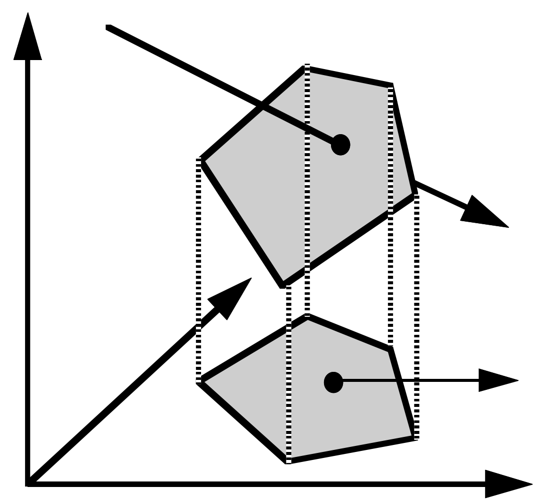
Solution

$$t = -\frac{ax_0 + by_0 + cz_0 + d}{a\Delta x + b\Delta y + c\Delta z}$$

If denominator = 0, ray is parallel to plane

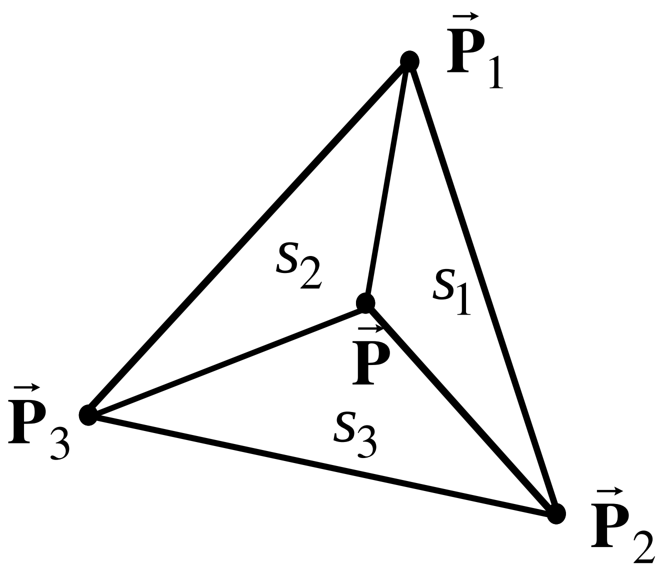
Ray-polygon intersection

1. Find intersection with plane of support
2. Test whether point is inside 3d polygon
   a. Project onto xy plane (actually drop largest of a,b,c in N)
   b. Test whether point is inside 2d polygon (use Jordan Curve Thm.)



(from Hanrahan)

Ray-triangle intersection



Barycentric coordinates

$$\vec{\mathbf{P}} = s_1\vec{\mathbf{P}}_1 + s_2\vec{\mathbf{P}}_2 + s_3\vec{\mathbf{P}}_3$$

$$s_1 + s_2 + s_3 = 1$$

Solution

$$s_1 = \frac{\left|\mathbf{P} \quad \mathbf{P}_2 \quad \mathbf{P}_3\right|}{\left|\mathbf{P}_1 \quad \mathbf{P}_2 \quad \mathbf{P}_3\right|}$$

$$s_2 = \frac{\left|\mathbf{P}_1 \quad \mathbf{P} \quad \mathbf{P}_3\right|}{\left|\mathbf{P}_1 \quad \mathbf{P}_2 \quad \mathbf{P}_3\right|}$$

$$s_3 = \frac{\left|\mathbf{P}_1 \quad \mathbf{P}_2 \quad \mathbf{P}\right|}{\left|\mathbf{P}_1 \quad \mathbf{P}_2 \quad \mathbf{P}_3\right|}$$

Inside criteria

$$0 \le s_1 \le 1$$

$$0 \le s_2 \le 1$$

$$0 \le s_3 \le 1$$

(from Hanrahan)

Other implicit surfaces

algebraic surfaces

- **implicit surface written as a polynomial of degree *n***
  - linear   plane
  - quadrics   sphere, cone, cylinder, paraboloid, hyperboloid
  - quartics   torus
  - superquadrics, etc.
- **substitute ray equation, then find roots as before**

tensor product surfaces

- **implicitization**
  - degree m in u, n in v $\rightarrow$ degree 2mn implicit surface
  - bilinear patch $\rightarrow$ degree 2 implicit surface
  - bicubic patch $\rightarrow$ degree 18 implicit surface

Procedural surfaces

tensor product surfaces (again)

- **hierarchy of bounding volumes**
  1. Fit bounding sphere around patch
  2. If ray intersects sphere, subdivide patch and recurse until “small”
  3. Approximate subpatch with rectangle

fractals

- **hierarchy of worst-case bounding prisms (“cheesecake extent”)**
  1. Fit bounding prism around triangle
  2. If ray intersects prism, subdivide triangle, displace, and recurse
  3. Intersect ray with triangle at lowest level of subdivision

height fields (a.k.a. displacement maps, a.k.a. terrains)

- **base plane with cells & heights**
    1. Project ray down to base plane
    2. Advance to next cell
    3. Test (3D) ray for intersection with terrain
- **hierarchy of bounding volumes is optional**

constructive solid geometry (CSG)

swept surfaces

deformed surfaces