8.1 Geometric Queries for Ray Tracing

Hao Li

http://cs420.hao-li.com

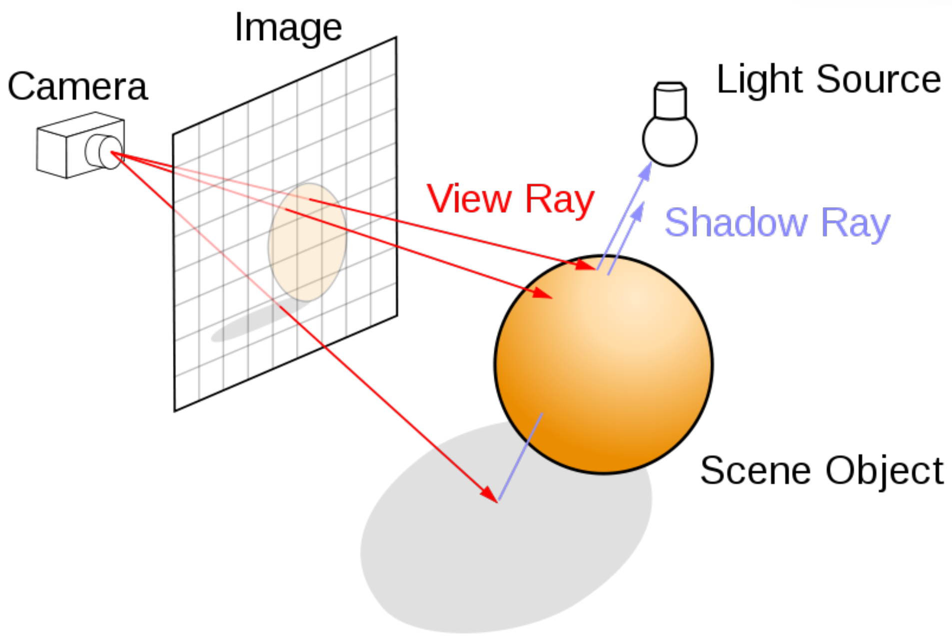
Outline

- Ray-Surface Intersections
- Special cases: sphere, polygon
- Barycentric coordinates

Outline

- **Ray-Surface Intersections**
- Special cases: sphere, polygon
- Barycentric coordinates

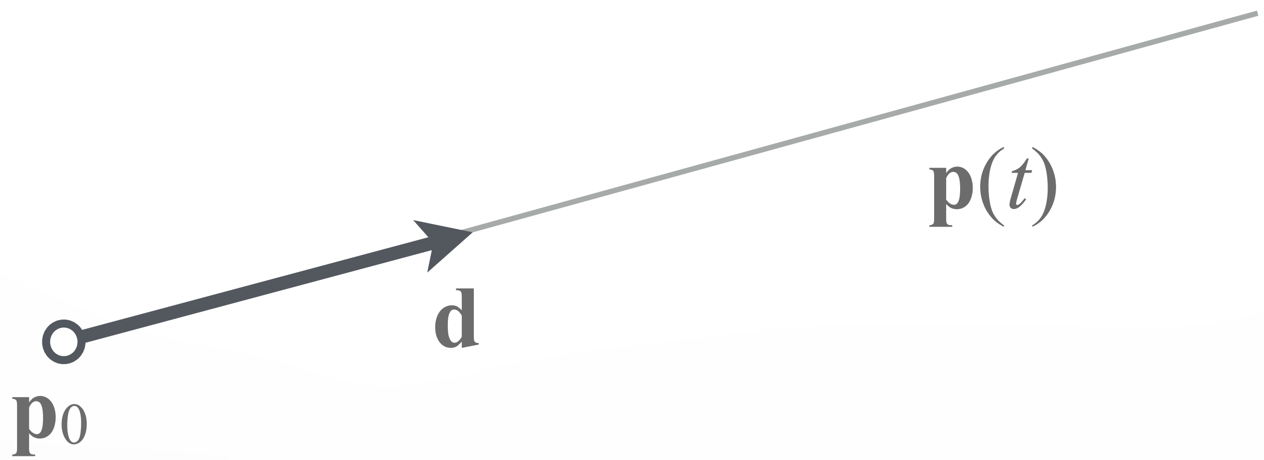
Ray-Surface Intersections



- Necessary in ray tracing
- General parametric surfaces
- General implicit surfaces
- Specialized analysis for special surfaces
  - Spheres
  - Planes
  - Polygons
  - Quadrics

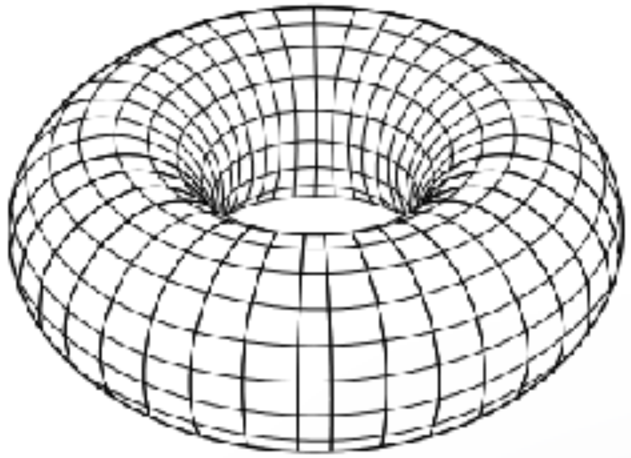
Generating Rays

- Ray in parametric form
  - Origin $\mathbf{p}_0 = [x_0 \;\; y_0 \;\; z_0]^T$
  - Direction $\mathbf{d} = [x_d \;\; y_d \;\; z_d]^T$
  - Assume $\mathbf{d}$ is normalized: $x_d \cdot x_d + y_d \cdot y_d + z_d \cdot z_d = 1$
  - Ray $\mathbf{p}(t) = \mathbf{p}_0 + \mathbf{d}t$ for $t > 0$



Intersection of Rays and Parametric Surfaces

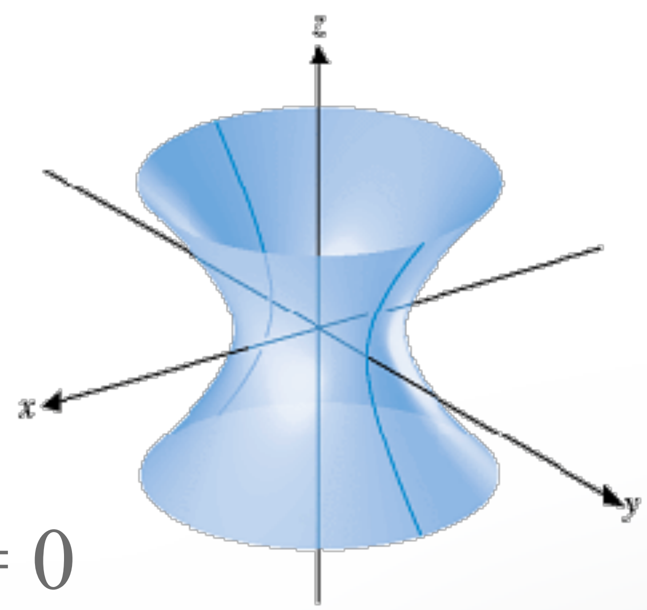
- Ray in parametric form
  - Origin $\mathbf{p}_0 = [x_0 \;\; y_0 \;\; z_0]^T$
  - Direction $\mathbf{d} = [x_d \;\; y_d \;\; z_d]^T$
  - Assume $\mathbf{d}$ is normalized: $x_d \cdot x_d + y_d \cdot y_d + z_d \cdot z_d = 1$
  - Ray $\mathbf{p}(t) = \mathbf{p}_0 + \mathbf{d}t$ for $t > 0$



- Surface in parametric form
  - Points $\mathbf{q} = \mathbf{g}(u, v) = [x(u, v), y(u, v), z(u, v)]$
  - Solve $\mathbf{p}_0 + \mathbf{d}t = \mathbf{g}(u, v)$
  - Three equations in three unknowns (t, u, v)
  - Possible bounds on u, v

Intersection of Rays and Implicit Surfaces

- Ray in parametric form
  - Origin $\mathbf{p}_0 = [x_0 \;\; y_0 \;\; z_0]^T$
  - Direction $\mathbf{d} = [x_d \;\; y_d \;\; z_d]^T$
  - Assume $\mathbf{d}$ is normalized: $x_d \cdot x_d + y_d \cdot y_d + z_d \cdot z_d = 1$
  - Ray $\mathbf{p}(t) = \mathbf{p}_0 + \mathbf{d}t$ for $t > 0$



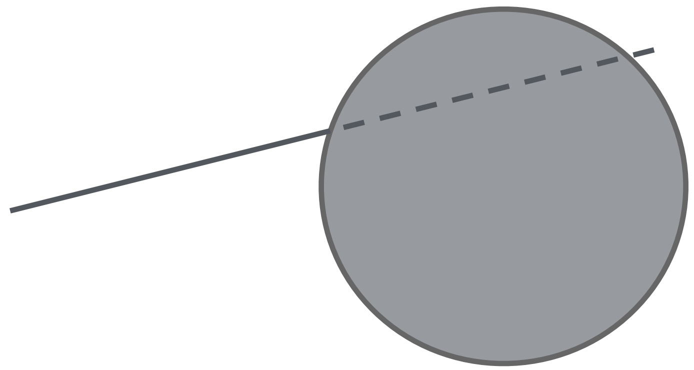
- Implicit surface
  - All points $\mathbf{q}$ such that $f(\mathbf{q}) = 0$
  - Substitute ray equation for $\mathbf{q}$: $f(\mathbf{p}_0 + \mathbf{d}t) = 0$
  - Solve for t (univariate root finding)
  - Closed form if possible, otherwise approximation

Outline

- Ray-Surface Intersections
- **Special cases: sphere, polygon**
- Barycentric coordinates

Ray-Sphere Intersection I

- Define sphere by
  - Center $\mathbf{c} = [x_c \;\; y_c \;\; z_c]^T$
  - Radius r
  - Implicit surface $f(\mathbf{q}) = (x - x_c)^2 + (y - y_c)^2 + (z - z_c)^2 - r^2 = 0$



- Plug in ray equations for x, y, z

$$x = x_0 + x_d\, t, \quad y = y_0 + y_d\, t, \quad z = z_0 + z_d\, t$$

- Obtain a scalar equation for t

$$(x_0 + x_d\, t - x_c)^2 + (y_0 + y_d\, t - y_c)^2 + (z_0 + z_d\, t - z_c)^2 - r^2 = 0$$

Ray-Sphere Intersection II

- Simplify to $at^2 + bt + c = 0$

  where $a = x_d^2 + y_d^2 + z_d^2 = 1 \quad \text{since } |d| = 1$

  $$b = 2(x_d(x_0 - x_c) + y_d(y_0 - y_c) + z_d(z_0 - z_c))$$

  $$c = (x_0 - x_c)^2 + (y_0 - y_c)^2 + (z_0 - z_c)^2 - r^2$$

- Solve to obtain t_0, t_1

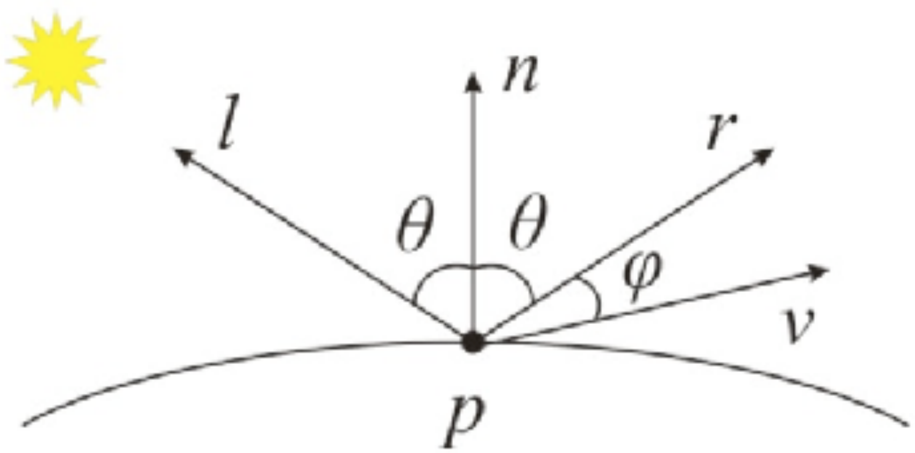
  $$t_{0,1} = \frac{-b \pm \sqrt{b^2 - 4ac}}{2}$$

- Check if $t_0, t_1 > 0$. Return $min(t_0, t_1)$

Ray-Sphere Intersection III

- For shading (e.g., Phong model), calculate unit normal

$$n = \frac{1}{r}[(x_i - x_c) \quad (y_i - y_c) \quad (z_i - z_c)]^T$$



- Negate if ray originates inside the sphere!

- Note possible problems with roundoff errors

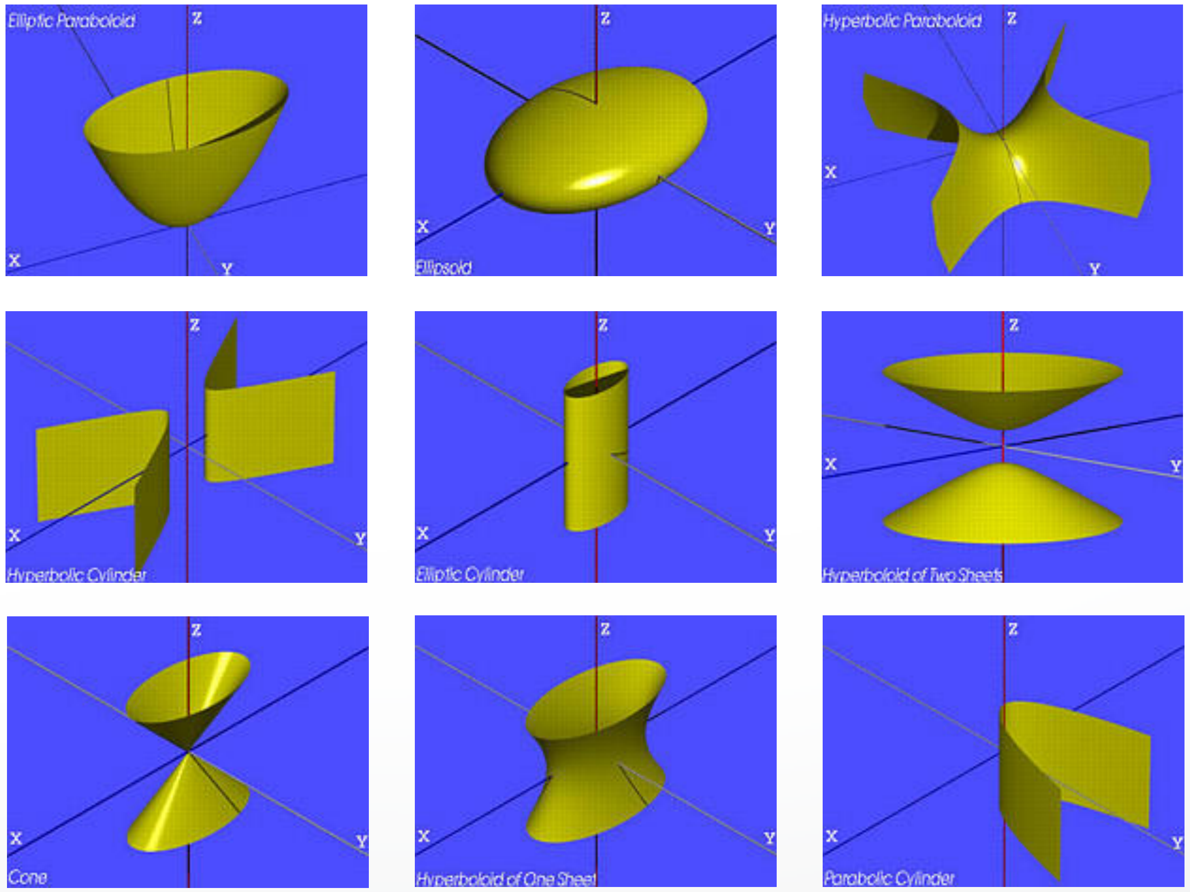
Simple Optimizations

- Factor common subexpressions

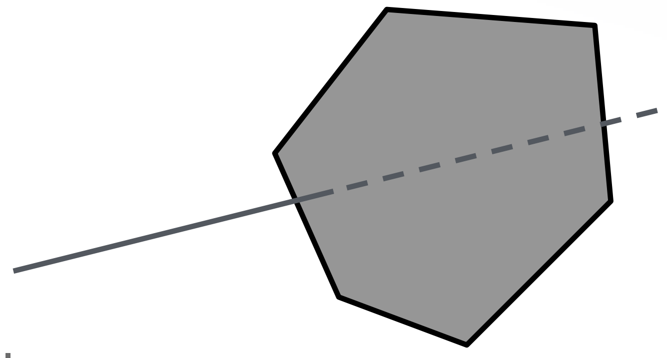
- Compute only what is necessary
  - Calculate $b^2 - 4ac$, abort if negative
  - Compute normal only for closest intersection
  - Other similar optimizations

Ray-Quadric Intersection

- Quadric $f(\mathbf{p}) = f(x, y, z) = 0$, where f is polynomial of order 2
  - Sphere, ellipsoid, paraboloid, hyperboloid, cone, cylinder
- Closed form solution as for sphere
- Combine with CSG



Ray-Polygon Intersection I



- Assume planar polygon in 3D
  1. Intersect ray with plane containing polygon
  2. Check if intersection point is inside polygon

- Plane
  - Implicit form: $a \cdot x + b \cdot y + c \cdot z + d = 0$
  - Unit normal: $\mathbf{n} = [a \;\; b \;\; c]^T$ with $a^2 + b^2 + c^2 = 1$

Ray-Polygon Intersection II

- Substitute t to obtain intersection point in plane

$$a(x_0 + x_d t) + b(y_0 + y_d t) + c(z_0 + z_d t) + d = 0$$

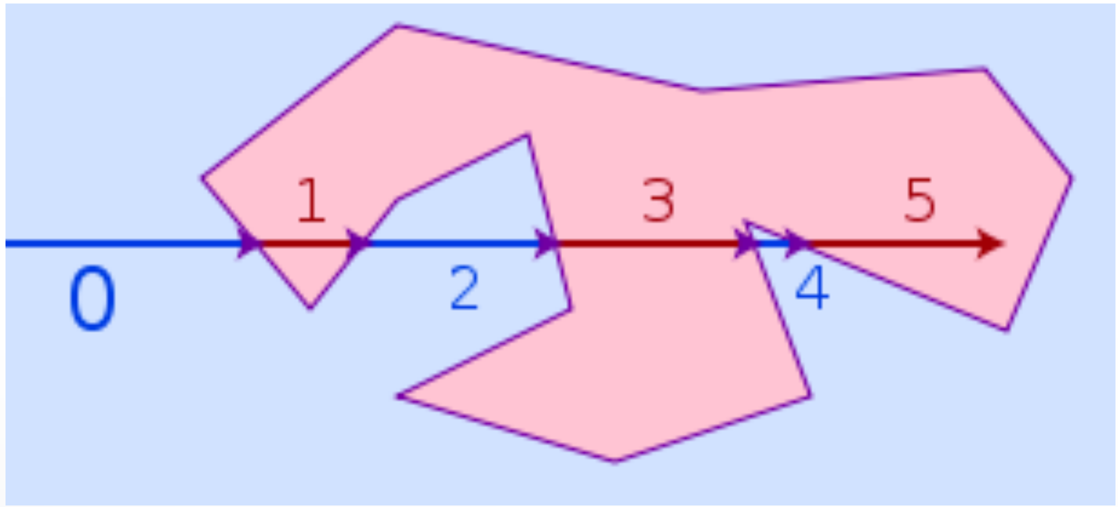
- Solve and rewrite using dot product

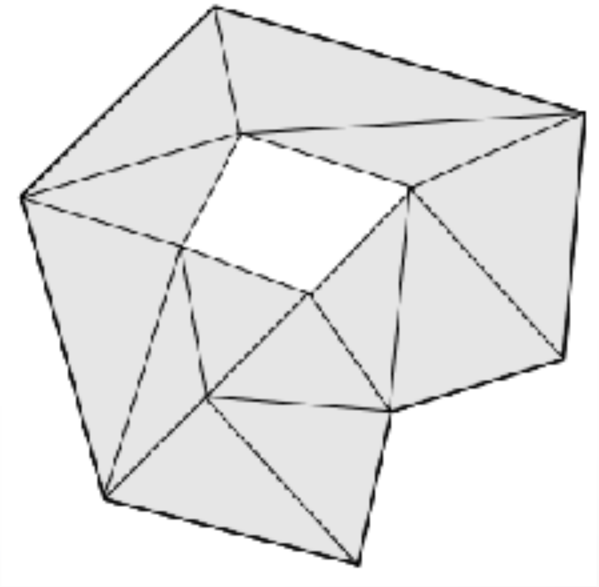
$$t = \frac{-(ax_0 + by_0 + cz_0 + d)}{ax_d + by_d + cz_d} = \frac{-(n \cdot p_0 + d)}{n \cdot d}$$

- If $n \cdot d = 0$, no intersection (ray parallel to plane)
- If $t \leq 0$, the intersection is behind ray origin

Test if point inside polygon

- Use even-odd rule or winding rule
- Easier if polygon is in 2D (project from 3D to 2D)
- Easier for triangles (tessellate polygons)





Point-in-triangle testing

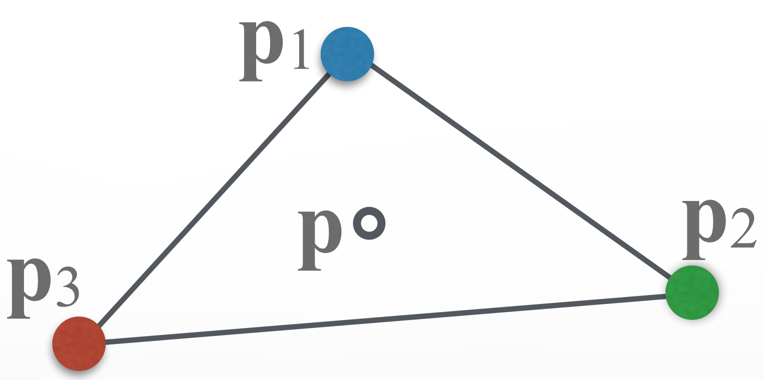
1. Project the point and triangle onto a plane
    - Pick a plane not perpendicular to triangle (such a choice always exists)
    - $x = 0$, $y = 0$, or $z = 0$
2. Then, do the 2D test in the plane, by computing barycentric coordinates (follows next)

Outline

- Ray-Surface Intersections
- Special cases: sphere, polygon
- **Barycentric coordinates**

Interpolated Shading for Ray Tracing

- Assume we know normals at vertices
- How do we compute normal of interior point?
- Need linear interpolation between 3 points
- Barycentric coordinates



Barycentric Coordinates in 1D

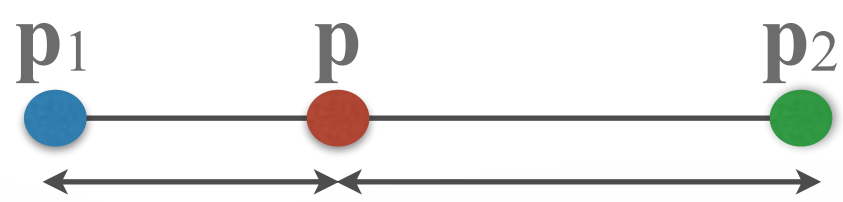
- Linear interpolation

$$\mathbf{p}(t) = (1 - t)\,\mathbf{p}_1 + t\,\mathbf{p}_2\ ,\ 0 \leq t \leq 1$$

$$\mathbf{p} = \alpha\,\mathbf{p}_1 + \beta\,\mathbf{p}_2\ ,\ \alpha + \beta = 1$$

$\mathbf{p}$ is between $\mathbf{p}_1$ and $\mathbf{p}_2$ *iff* $0 \leq \alpha, \beta \leq 1$

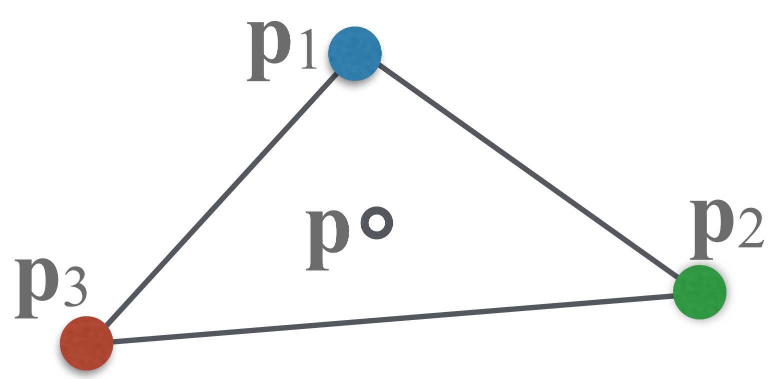
- Geometric intuition
  - Weigh each vertex by ratio of distances from ends



- α, β are called barycentric coordinates

Barycentric Coordinates in 2D

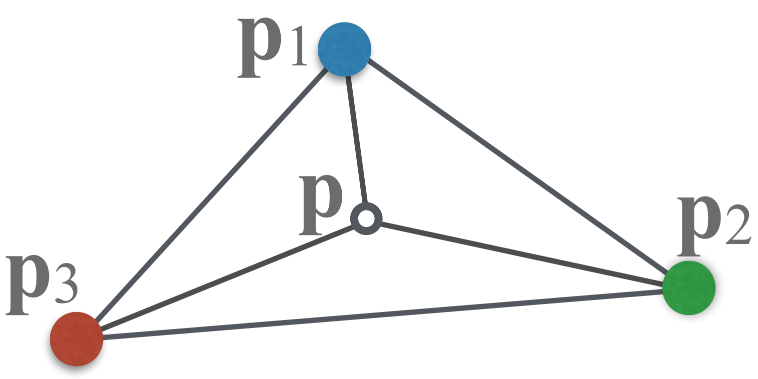
- Now we have 3 points instead of 2



- Define 3 barycentric coordinates α, β, γ
- $\mathbf{p} = \alpha\, \mathbf{p}_1 + \beta\, \mathbf{p}_2 + \gamma\, \mathbf{p}_3$
- $\mathbf{p}$ inside triangle *iff* $0 \leq \alpha, \beta, \gamma \leq 1$, $\alpha + \beta + \gamma = 1$
- How do we calculate α, β, γ?

Barycentric Coordinates for Triangle

- Coordinates are ratios of triangle areas



$$\alpha = \text{Area}(\mathbf{p}\mathbf{p}_2\mathbf{p}_3) \,/\, \text{Area}(\mathbf{p}_1\mathbf{p}_2\mathbf{p}_3)$$

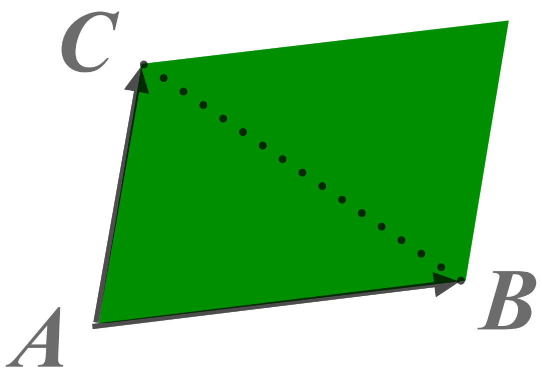
$$\beta = \text{Area}(\mathbf{p}_1\mathbf{p}\mathbf{p}_3) \,/\, \text{Area}(\mathbf{p}_1\mathbf{p}_2\mathbf{p}_3)$$

$$\gamma = \text{Area}(\mathbf{p}_1\mathbf{p}_2\mathbf{p}) \,/\, \text{Area}(\mathbf{p}_1\mathbf{p}_2\mathbf{p}_3) = 1 - \alpha - \beta$$

- Areas in these formulas should be signed
  - Clockwise (-) or anti-clockwise (+) orientation of the triangle
  - Important for point-in-triangle test

Compute Triangle Area in 3D

- Use cross product
- Parallelogram formula
- Area($\boldsymbol{ABC}$) = (1/2) $|(\boldsymbol{B} - \boldsymbol{A}) \times (\boldsymbol{C} - \boldsymbol{A})|$
- How to get correct sign for barycentric coordinates?
  - Compare directions of cross product $(\boldsymbol{B} - \boldsymbol{A}) \times (\boldsymbol{C} - \boldsymbol{A})$ for triangles $\mathbf{p}\mathbf{p}_2\mathbf{p}_3$ vs $\mathbf{p}_1\mathbf{p}_2\mathbf{p}_3$, etc. (either 0 (sign+) or 180 deg (sign-) angle)
  - Easier alternative: project to 2D, use 2D formula (projection to 2D preserves barycentric coordinates)



Compute Triangle Area in 2D

- Suppose we project the triangle ***ABC*** to x-y plane
- Area of the projected triangle in 2D with the correct sign:

$$(1/2)((b_x - a_x)(c_y - a_y) - (c_x - a_x)(b_y - a_y))$$

Outline

- Ray-Surface Intersections
- Special cases: sphere, polygon
- Barycentric coordinates

http://cs420.hao-li.com

Thanks!