Ray Tracing 2

Week 12

CS5600 ***Computer Graphics***

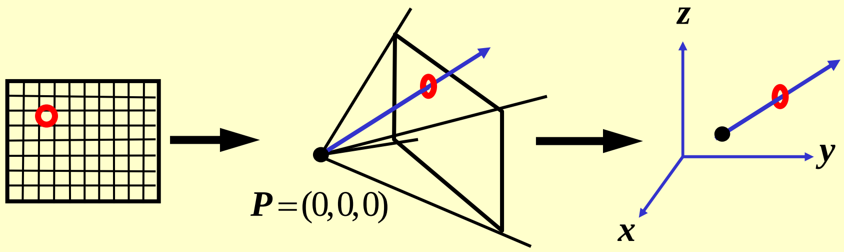
From Rich Riesenfeld

Spring 2009

Ray Tracing

- Classical geometric optics technique
- Extremely versatile
- Historically viewed as expensive
- Good for special effects
- Computationally intensive
- Can do sophisticated graphics

Ray Tracing

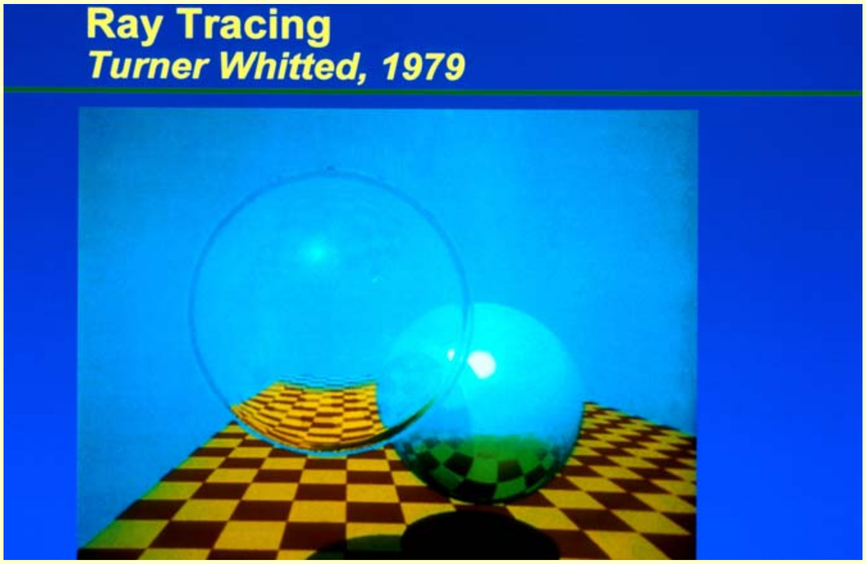


Screen-space point

Film plane point in canonical view volume

World-space ray

What is the Projection Matrix?



Environment Mapping

Ray Tracing Implementation - 1

- Key computation: Must find

  $$ray \cap object$$

- This is equivalent to

  $$ray - object = 0$$

- This is essentially *root finding*

Ray Tracing Implementation - 2

- Ray is often represented parametrically,

  $$\boldsymbol{r}(t) = t\,(\boldsymbol{P} - \boldsymbol{E})\ ,$$

  so we seek,

  $$\boldsymbol{r}(t) \cap \boldsymbol{F}(x,y,z)$$

- Problem requires intersection of parametric ray with some kind of surface
- Ray Tracing maps easily onto recursion

Ray Tracing Implementation

- 3

- RT’ing used for spectacular images
- RT’ing maps naturally to recursion
- RT’ing is trivially parallelized
- RT’ing has robustness problems
- RT’ing has aliasing problems

Ray Tracing

- 1

Three (*nonexclusive*) phenomena follow when ray intersects object:

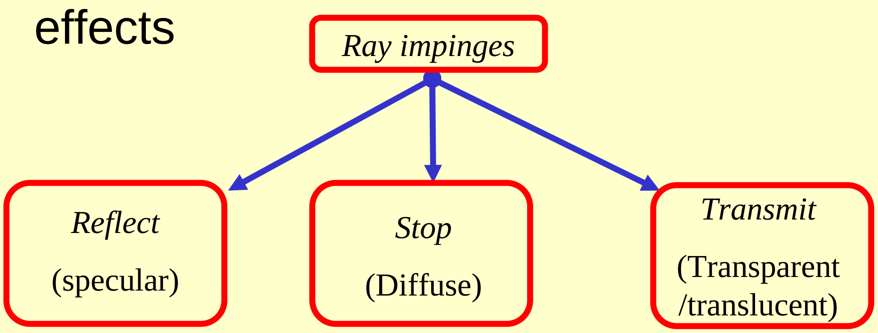
1. *Reflect* (specularity)
2. *Pass through* (transparency)
3. *Stop* (diffuse - look for light vector and calculate proper value)

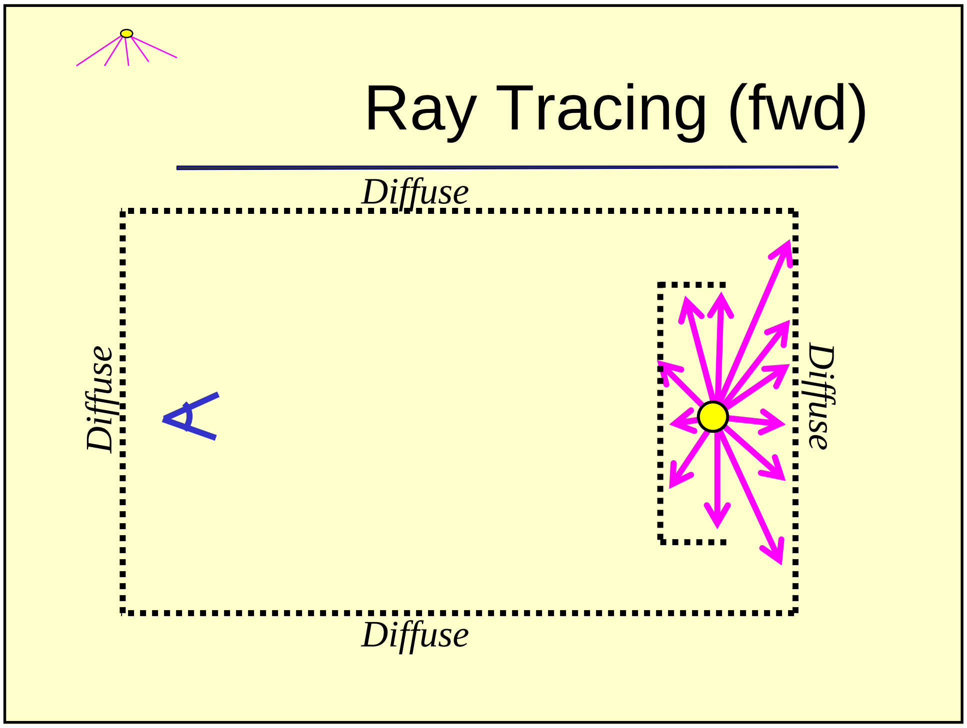
Ray Tracing

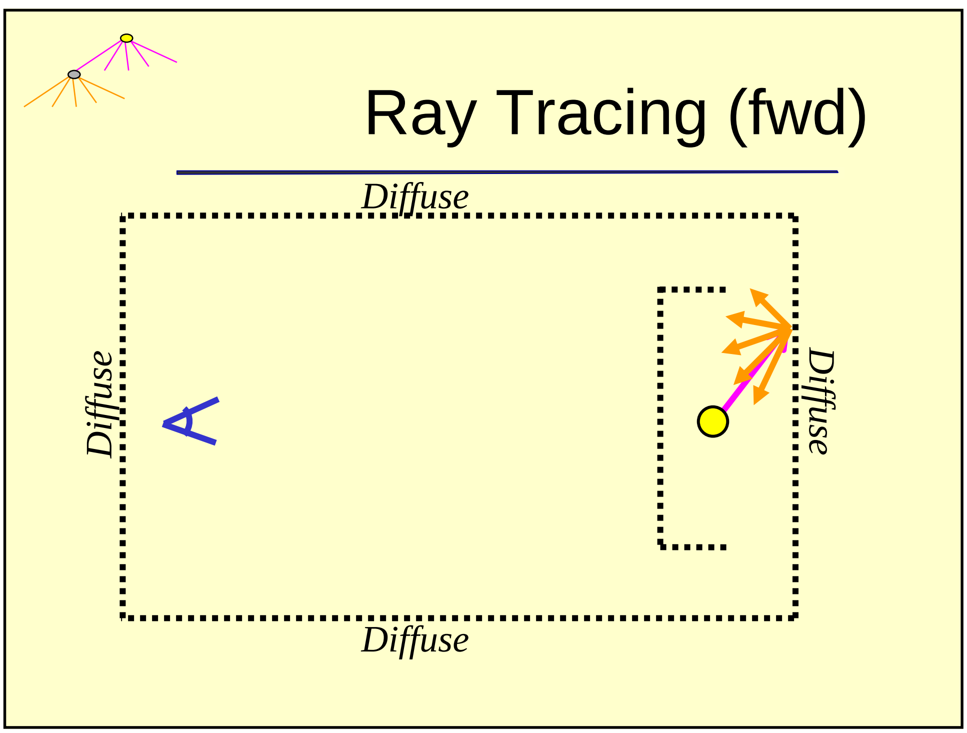
- Forward ray tracing: $E(S*)DL$
- Backward ray tracing: $L(S*)DE$
- What is the difference?

Ray Tracing (ternary) Tree

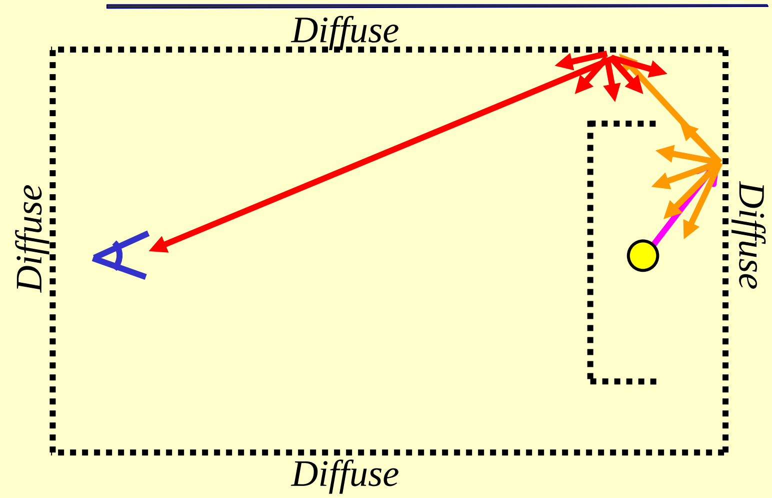
Often a combination of all three occur at each node to model sophisticated effects



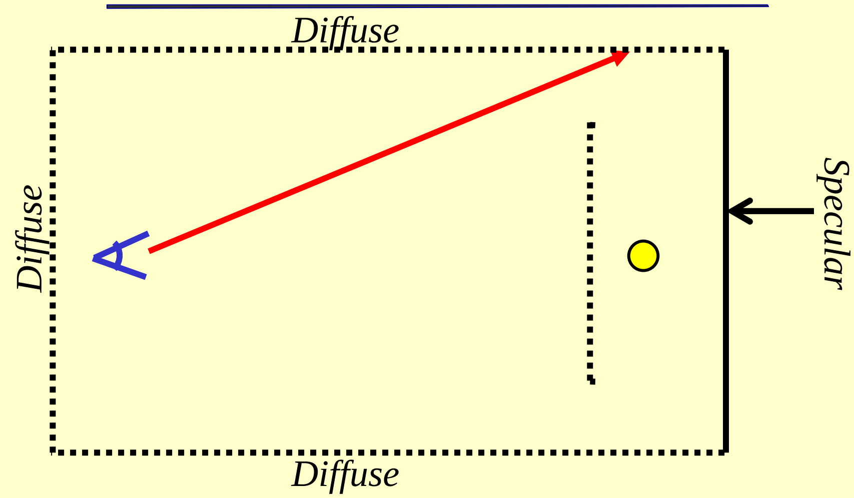

Ray Tracing (fwd)
Diffuse
Diffuse
Diffuse
Diffuse


Ray Tracing (fwd)
Diffuse
Diffuse
Diffuse
Diffuse

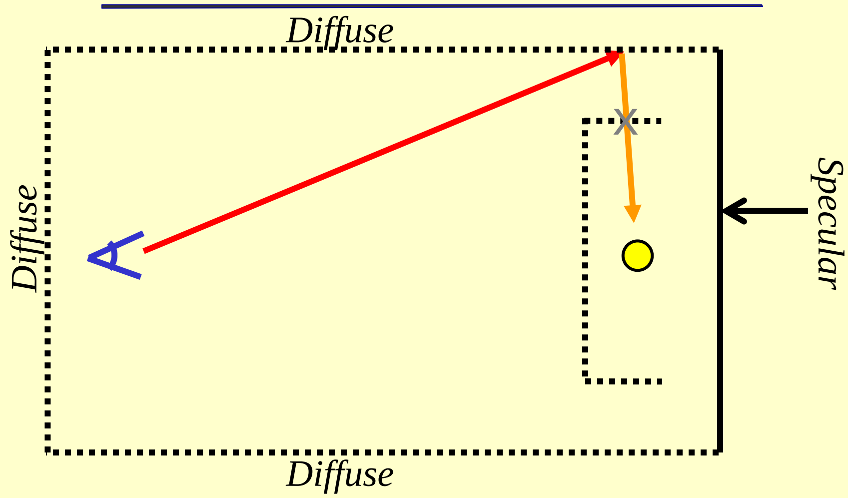
Ray Tracing (fwd)



RT'ing Indirect Light (bkwd)

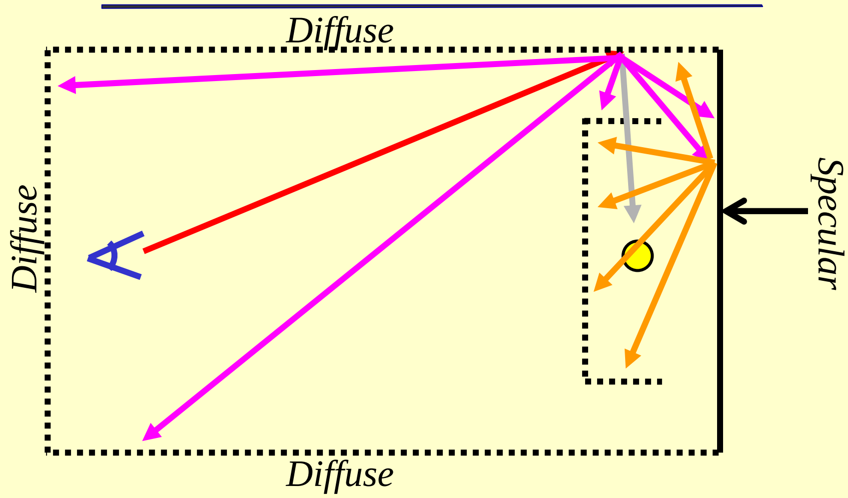


RT’ing Indirect Light (bkwd)



Spring 2006 CS5600 17

RT’ing Indirect Light (bkwd)

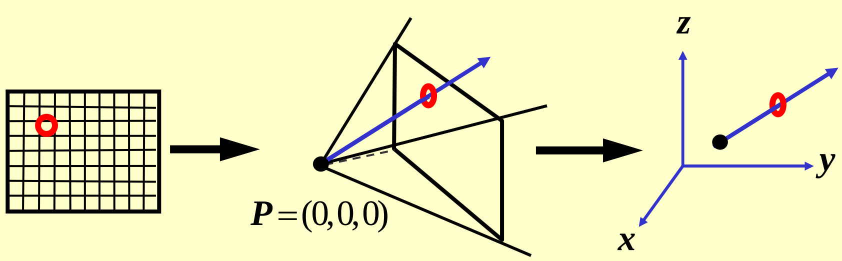


Spring 2006 CS5600 18

Ray Tracing Growth

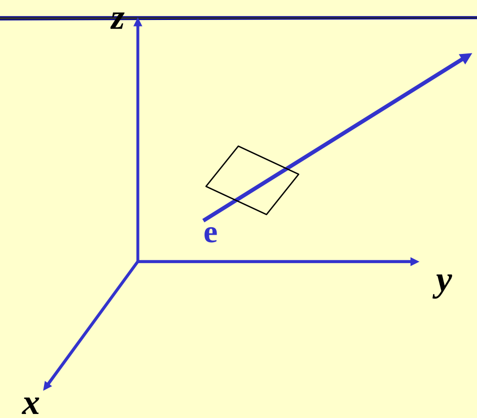
- Tree can grow extremely fast, with high exponential fan out
- Fancy rays can have many cross-section geometries; not necessary a line with 0-cross-section
- Need to bound tree depth & fanout

Returning to Ray Tracing



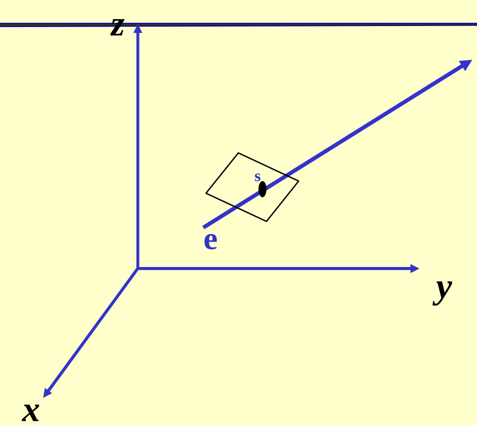
Screen-space point
Film plane point in canonical view volume
World-space ray

Returning to Ray Tracing



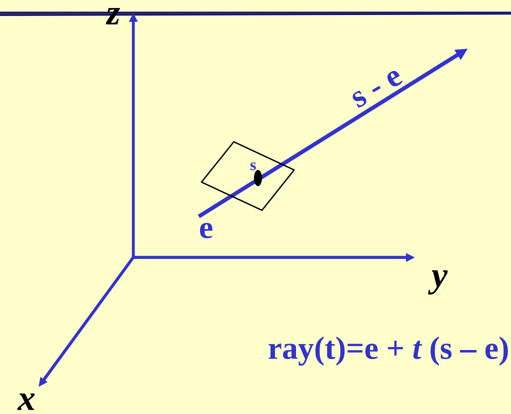
World-space ray

Returning to Ray Tracing



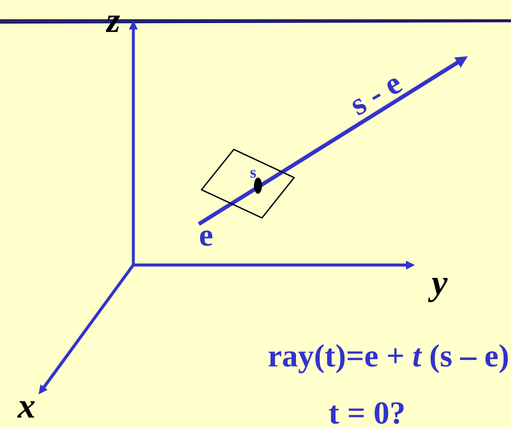
World-space ray

Returning to Ray Tracing



World-space ray

Returning to Ray Tracing



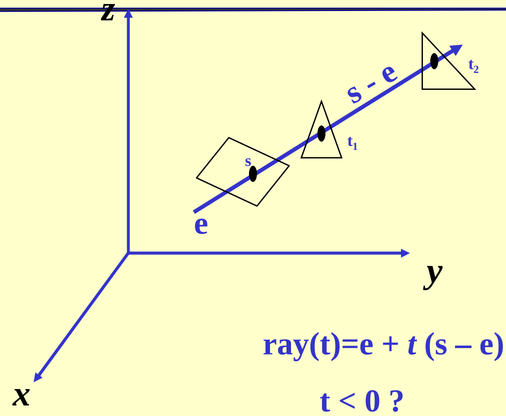
t = 0?

World-space ray

t = s?

t < 0 ?

Returning to Ray Tracing



ray(t)=e + *t* (s – e)

t < 0 ?

World-space ray

$t_1 < t_2$

Returning to Ray Tracing

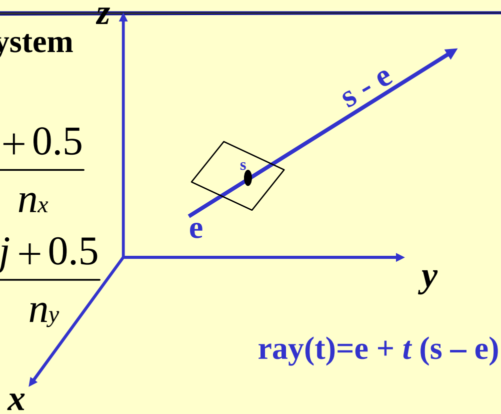
***uvw* coordinate system**

Origin is e

$$u_s = l + (r - l)\frac{i + 0.5}{n_x}$$

$$v_s = b + (t - b)\frac{j + 0.5}{n_y}$$

$$w_s = near$$



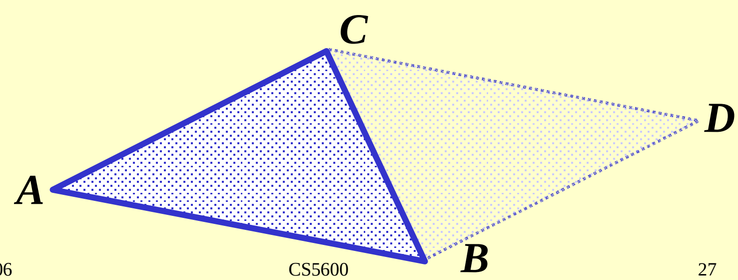
ray(t)=e + *t* (s – e)

World-space ray

Computing Areas: Recall

$$Area\ \square\, \boldsymbol{ABDC} = \boldsymbol{AB} \times \boldsymbol{AC}$$

$$Area\ \square\, \boldsymbol{ABC} = \tfrac{1}{2}\boldsymbol{AB} \times \boldsymbol{AC}$$

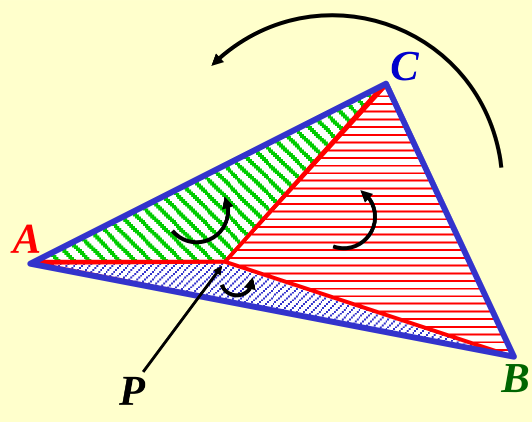


Barycentric Coords: Areas

$$\alpha = \frac{Area\ \square\, \boldsymbol{PBC}}{Area\ \square\, \boldsymbol{ABC}}$$

$$\beta = \frac{Area\ \square\, \boldsymbol{PCA}}{Area\ \square\, \boldsymbol{ABC}}$$

$$\gamma = \frac{Area\ \square\, \boldsymbol{PAB}}{Area\ \square\, \boldsymbol{ABC}}$$



Barycentric Coords: Areas

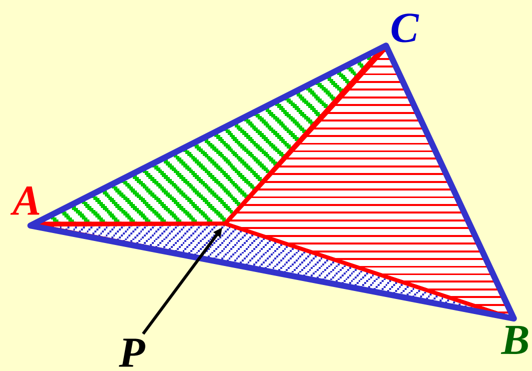
Then,

$\alpha, \beta, \gamma \geq 0,$

$\alpha + \beta + \gamma \equiv 1,$

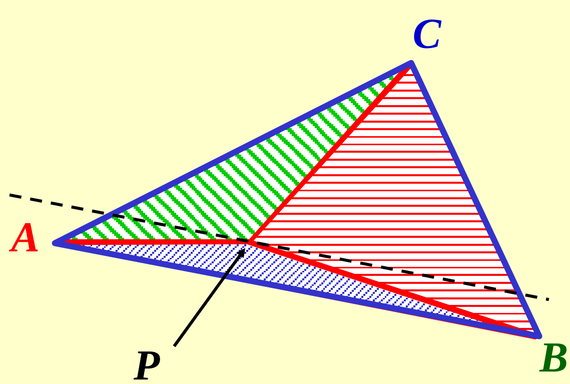
and, $P =$

$\alpha A + \beta B + \gamma C$



Isoparametric lines

constant γ



Extended Barycentric Coords

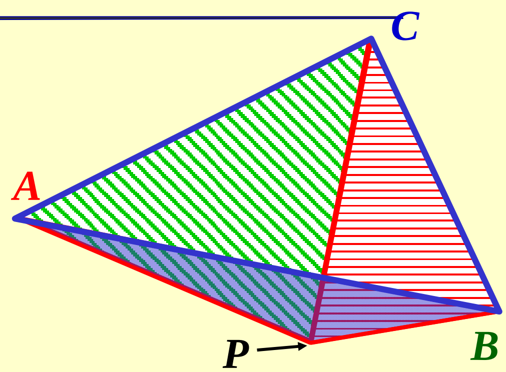
Then,

~~$\alpha,\beta,\gamma \ge 0$,~~

$\alpha + \beta + \gamma \equiv 1$,

and, $\boldsymbol{P} =$

$\alpha \boldsymbol{A} + \beta \boldsymbol{B} + \gamma \boldsymbol{C}$



P can be *outside* triangle if coord's allowed to go negative.

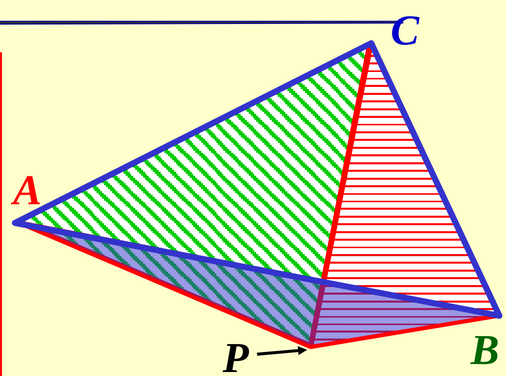
Spring 2006 CS5600 31

Extended Barycentric Coords

Can use

$\alpha,\beta,\gamma \le 0$?

for test of ***P*** outside □ ***ABC*** ?

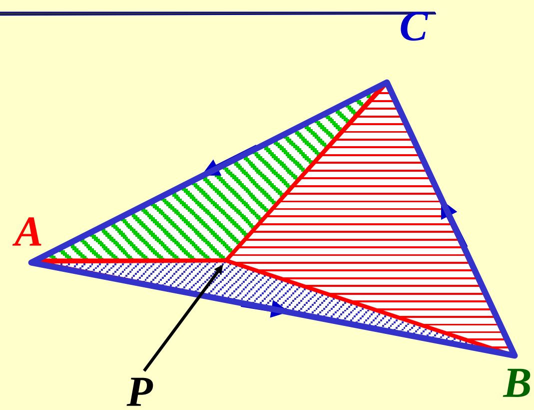


P can be *outside* triangle, but in it plane, if coord's can be negative.

Spring 2006 CS5600 32

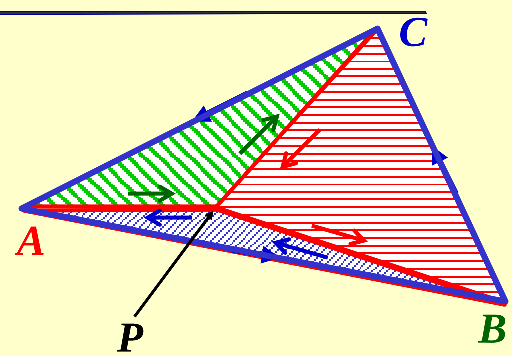
Algorithm for Computing Barycentric Coords of *P* - 1

1. As indicated, assign an orientation to □***ABC***

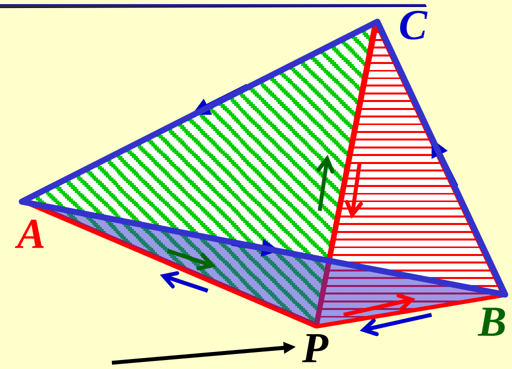


Algorithm for Computing Barycentric Coords of *P* - 2

2. Compute all areas using □***PAB***,□***PBC*** and □***PCA*** Orientation is important!



Algorithm for Computing Barycentric Coords of *P* - 3



Note this works equally for *P outside*

Algorithm for Computing Barycentric Coords of *P* - 4

3. Compute all areas using cross products in the following manner,

$$2*Area_{big} = \|\mathbf{AB} \times \mathbf{BC}\|$$
$$2*Area_{\alpha} = \|\mathbf{PB} \times \mathbf{BC}\|$$
$$2*Area_{\beta} = \|\mathbf{PC} \times \mathbf{CA}\|$$
$$2*Area_{\gamma} = \|\mathbf{PA} \times \mathbf{AB}\|$$

Algorithm for Computing Barycentric Coords of *P* - 5

4. Now compute ratios for α, β, γ

$$\alpha = \frac{2*Area_{\alpha}}{2*Area_{big}} = \frac{\|\mathbf{PB}\times\mathbf{BC}\|}{\|\mathbf{AB}\times\mathbf{BC}\|}$$

$$\beta = \frac{2*Area_{\alpha}}{2*Area_{big}} = \frac{\|\mathbf{PC}\times\mathbf{CA}\|}{\|\mathbf{AB}\times\mathbf{BC}\|}$$

$$\gamma = \frac{2*Area_{\alpha}}{2*Area_{big}} = \frac{\|\mathbf{PA}\times\mathbf{AC}\|}{\|\mathbf{AB}\times\mathbf{BC}\|}$$

Algorithm for Computing Barycentric Coords of *P* - 6

5. Now compute sign of area. *In* or *out*? Look at *dot product* of area vectors! *P* is *in* if same *sign*, i.e., positive; *out*, outwise.

$$Sign_{\alpha} = sign\{(\mathbf{PB}\times\mathbf{BC})\cdot(\mathbf{AB}\times\mathbf{BC})\}$$

$$Sign_{\beta} = sign\{(\mathbf{PC}\times\mathbf{CB})\cdot(\mathbf{AB}\times\mathbf{BC})\}$$

$$Sign_{\gamma} = sign\{(\mathbf{PA}\times\mathbf{AC})\cdot(\mathbf{AB}\times\mathbf{BC})\}$$

Applet

Universität Karlsruhe (TH)
Geometrische Datenverarbeitung:

<http://i33www.ira.uka.de/applets/mocca/html/noplugin/inhalt.html>

Spring 2006 CS5600 39

Ray Intersect Parametic Object

Let $\boldsymbol{S}(u,v) = \left(f(u,v), g(u,v), h(u,v) \right)$

be a parametrically defined object.

Componentwise, this means,

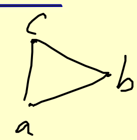
$$e_x + td_x = f(u,v)$$

$$e_y + td_y = g(u,v)$$

$$e_z + td_z = h(u,v)$$

Spring 2006 CS5600 40

Ray Triangle



$$e + td = a + \beta(b-a) + \gamma(c-a)$$

some

$$x_e + tx_d = x_a + \beta(x_b - x_a) + \gamma(x_c - x_a)$$

$$y_e + ty_d = y_a + \beta(y_b - y_a) + \gamma(y_c - y_a)$$

$$z_e + tz_d = z_a + \beta(z_b - z_a) + \gamma(z_c - z_a)$$

Ray Triangle

Cramer's M X Y

$$\begin{bmatrix} x_a - x_b & x_a - x_c & x_d \\ y_a - y_b & y_a - y_c & y_d \\ z_a - z_b & z_a - z_c & z_d \end{bmatrix} \begin{bmatrix} \beta \\ \gamma \\ t \end{bmatrix} \begin{matrix} X_1 \\ X_2 \\ X_3 \end{matrix} = \begin{bmatrix} x_a - x_e \\ y_a - y_e \\ z_a - z_e \end{bmatrix}$$

Recall Cramer's Rule - 1

Let, $$\begin{bmatrix} m_{11} & m_{12} & m_{13} \\ m_{21} & m_{22} & m_{23} \\ m_{31} & m_{32} & m_{33} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} y_1 \\ y_2 \\ y_3 \end{bmatrix}$$

That is, $\boldsymbol{Mx} = \boldsymbol{y}$

and, $$\|\boldsymbol{M}\| = \begin{Vmatrix} m_{11} & m_{12} & m_{13} \\ m_{21} & m_{22} & m_{23} \\ m_{31} & m_{32} & m_{33} \end{Vmatrix} = \det(\boldsymbol{M})$$

Spring 2006 CS5600 43

Recall Cramer's Rule - 2

Then,

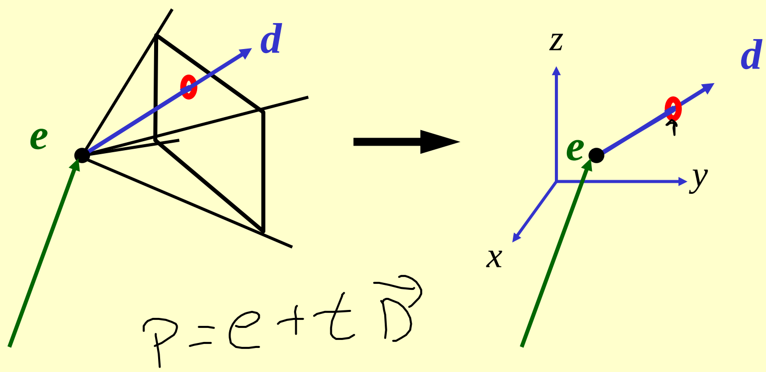
$$\beta = x_1 = \frac{1}{\|\boldsymbol{M}\|} \begin{Vmatrix} y_1 & m_{12} & m_{13} \\ y_2 & m_{22} & m_{23} \\ y_3 & m_{32} & m_{33} \end{Vmatrix}; \quad \gamma = x_2 = \frac{1}{\|\boldsymbol{M}\|} \begin{Vmatrix} m_{11} & y_1 & m_{13} \\ m_{21} & y_2 & m_{23} \\ m_{31} & y_3 & m_{33} \end{Vmatrix};$$

and,

$$t = x_3 = \frac{1}{\|\boldsymbol{M}\|} \begin{Vmatrix} m_{11} & m_{12} & y_1 \\ m_{21} & m_{22} & y_2 \\ m_{31} & m_{32} & y_3 \end{Vmatrix}$$

Spring 2006 CS5600 44

Ray Tracing



Barycentric Coords: Areas

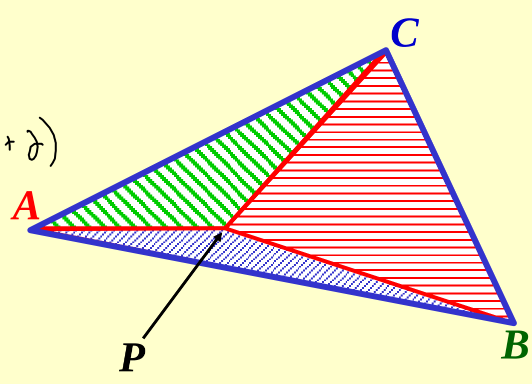
Then,

$\alpha, \beta, \gamma \geq 0,$

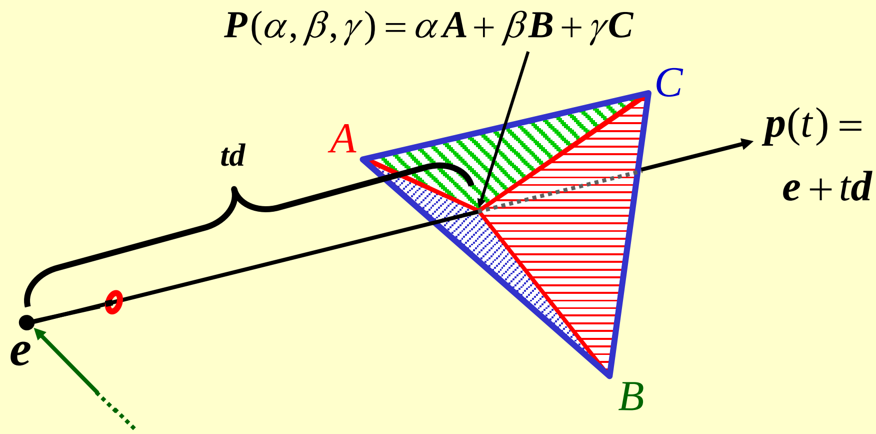
$\alpha + \beta + \gamma \equiv 1,$ $-(\beta+\gamma)$

and, $\boldsymbol{P} =$

$\alpha \boldsymbol{A} + \beta \boldsymbol{B} + \gamma \boldsymbol{C}$



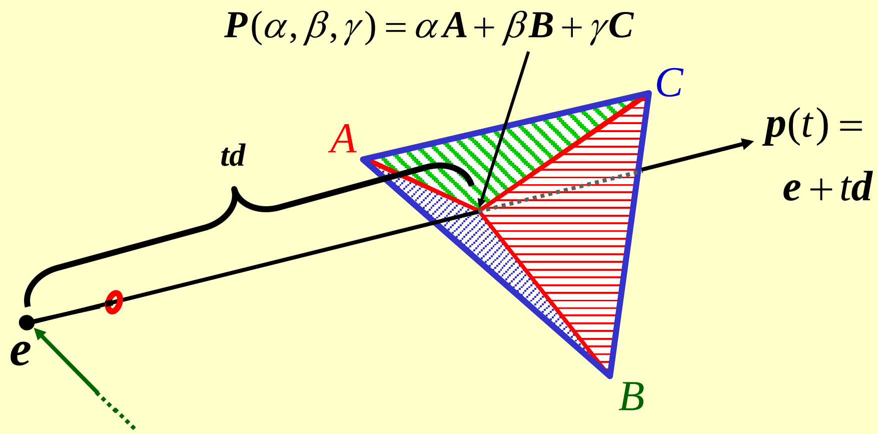
Solving for $\boldsymbol{p}(t) \cap \triangle \boldsymbol{ABC}$



What about texturing?

We know the intersection point, *p*, what about (*s,t*)?

Solving for $\boldsymbol{p}(t) \cap \triangle \boldsymbol{ABC}$



Shirley's $ray \cap$ Polygon Method

Let $\boldsymbol{p}(t) = \boldsymbol{e} + t\boldsymbol{d}$ be a ray.

For any object defined by f,

we are looking for

$$\mathrm{f}\left(\boldsymbol{p}(t)\right) = \mathrm{f}\left(\boldsymbol{e} + t\boldsymbol{d}\right) = 0,$$

for intersection points

Shirley's $ray \cap$ Polygon Method

$f(p(t)) = f(e + t\mathbf{d}) = 0$

Equation to test point in plane:

$$(p - p_1) \text{ dot } n = 0$$

Plug ray-equ into plane-equ:

$$(e + t\mathbf{d} - p_1) \text{ dot } n = 0$$

Shirley's $ray \cap$ Polygon Method

$$(e + t\mathbf{d} - p_1) \text{ dot } n = 0$$

$$(t\mathbf{d} + e - p_1) \text{ dot } n = 0$$

$$(t\mathbf{d} \text{ dot } n) + (e - p_1) \text{ dot } n = 0$$

$$t\mathbf{d} \text{ dot } n = (p_1 - e) \text{ dot } n$$

$$t = (p_1 - e) \text{ dot } n \,/\, \mathbf{d} \text{ dot } n$$

Need to check if p in inside polygon

$ray \cap implicit\_object$

Let $\boldsymbol{p}(t) = \boldsymbol{e} + t\boldsymbol{d}$ be a ray.

For any object defined by

$f(x, y, z) = 0$, we are looking for

$f\left(\boldsymbol{p}(t)\right) = f\left(\boldsymbol{e} + t\boldsymbol{d}\right) = 0,$

for intersection points

$ray \cap sphere$

- 1

For sphere with center $\boldsymbol{c} = (c_x, c_y, c_z)$,

$(\boldsymbol{p} - \boldsymbol{c}) \bullet (\boldsymbol{p} - \boldsymbol{c}) - R^2 = 0$

is a vector eq with $\boldsymbol{p} = (p_x, p_y, p_z)$.

$\therefore \quad (\boldsymbol{e} + t\boldsymbol{d} - \boldsymbol{c}) \bullet (\boldsymbol{e} + t\boldsymbol{d} - \boldsymbol{c}) - R^2 = 0$

ray ∩ sphere

- 2

Collecting terms,

$$\mathbf{d}\bullet\mathbf{d}\, t^2 + 2\mathbf{d}\bullet(\mathbf{e}-\mathbf{c})t + (\mathbf{e}-\mathbf{c})\bullet(\mathbf{e}-\mathbf{c}) - R^2 = 0$$

Hence, $t =$

$$\frac{-\mathbf{d}\bullet(\mathbf{e}-\mathbf{c}) \pm \sqrt{\left(\mathbf{d}\bullet(\mathbf{e}-\mathbf{c})\right)^2 - (\mathbf{d}\bullet\mathbf{d})\left((\mathbf{e}-\mathbf{c})\bullet(\mathbf{e}-\mathbf{c}) - R^2\right)}}{(\mathbf{d}\bullet\mathbf{d})}$$

And, unit normal is $\mathbf{n} = {(\mathbf{p}-\mathbf{c})}/{R}$

ray ∩ sphere

Sign of: $\left(\mathbf{d}\bullet(\mathbf{e}-\mathbf{c})\right)^2 - (\mathbf{d}\bullet\mathbf{d})\left((\mathbf{e}-\mathbf{c})\bullet(\mathbf{e}-\mathbf{c}) - R^2\right)$

if sign < 0 : ray misses sphere
sign = 0: ray is tangent to sphere
sign > 0: 2 intersections

Which to use?

Lecture Week 12 B

End

Ray Tracing B