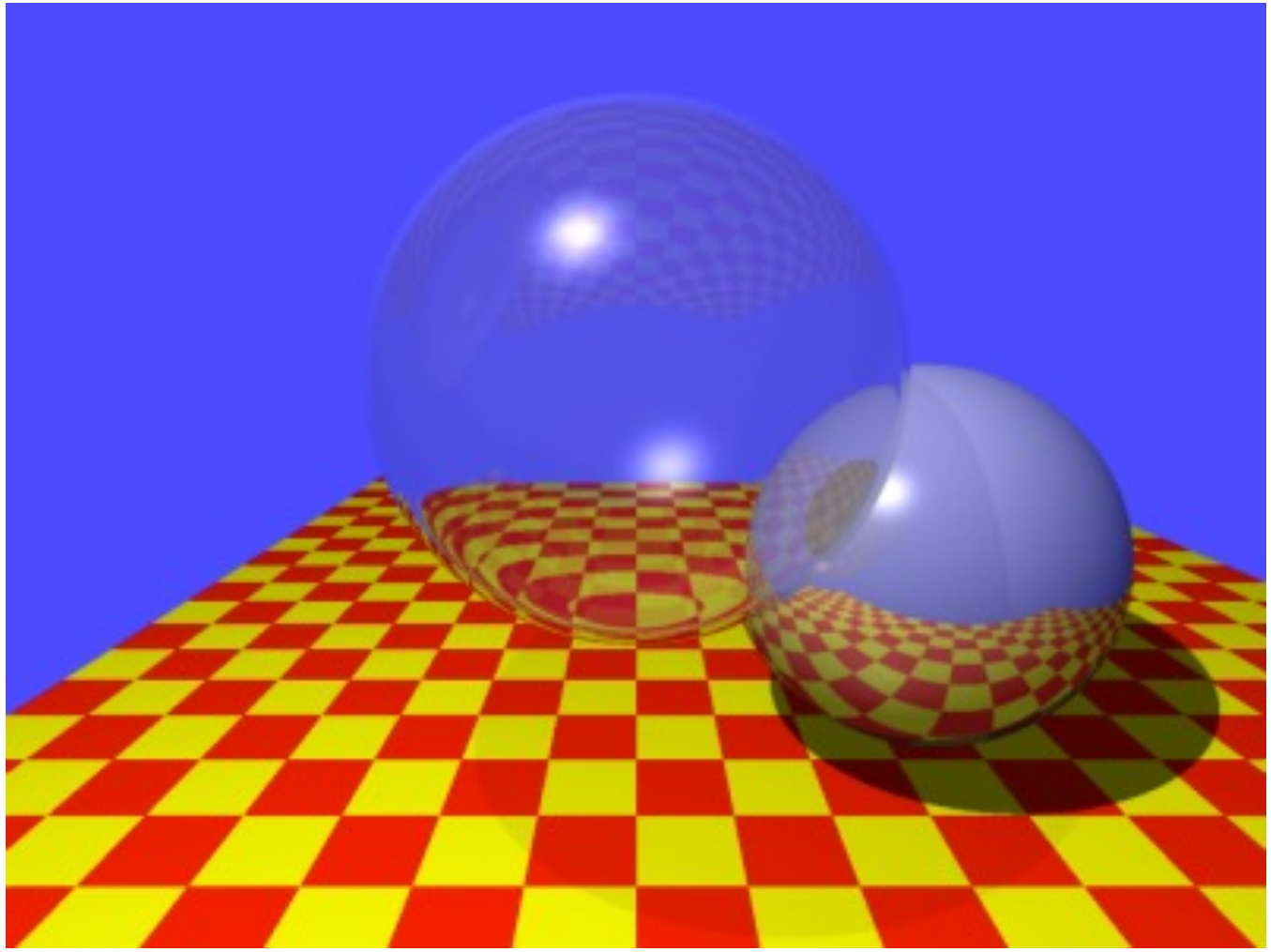
13 – raytracing (1)

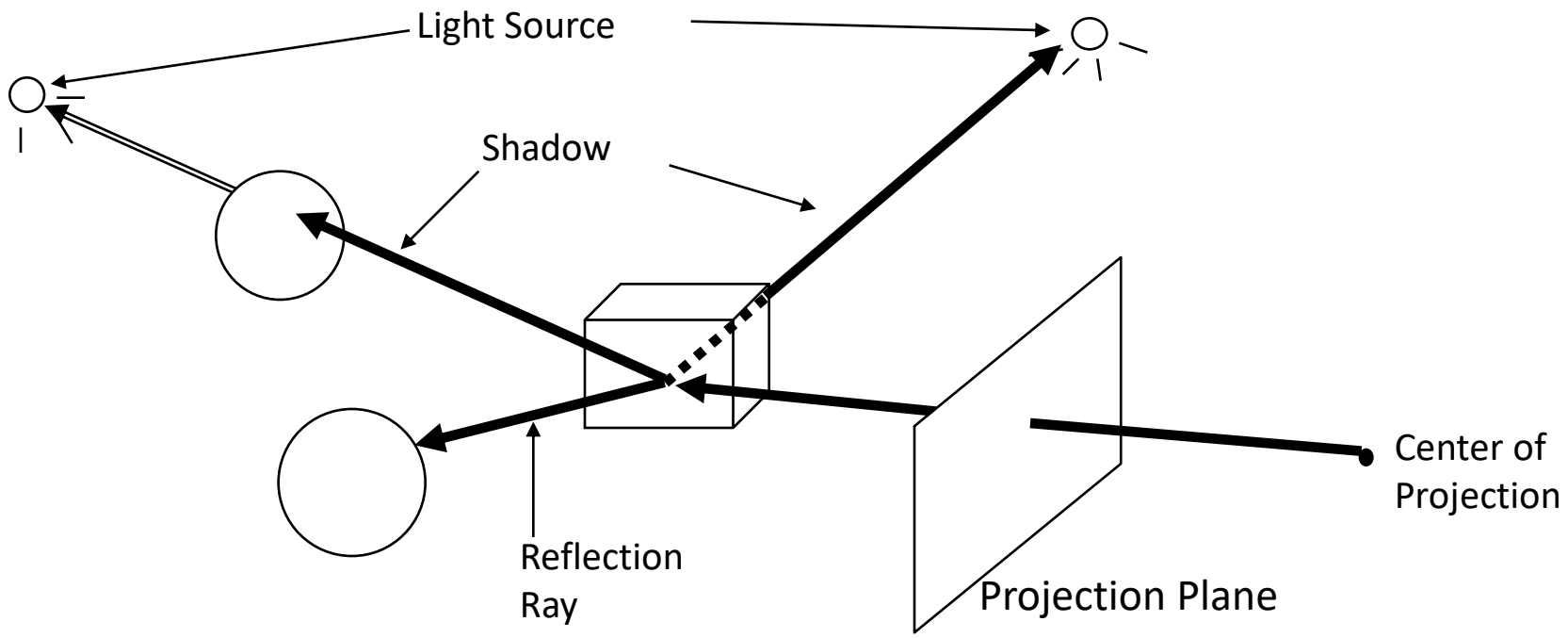
3 approaches to graphics

- On-line / “real-time”
  - Immediate mode
  - Retained mode
- Off-line / batch / “slow”

Ray Tracing



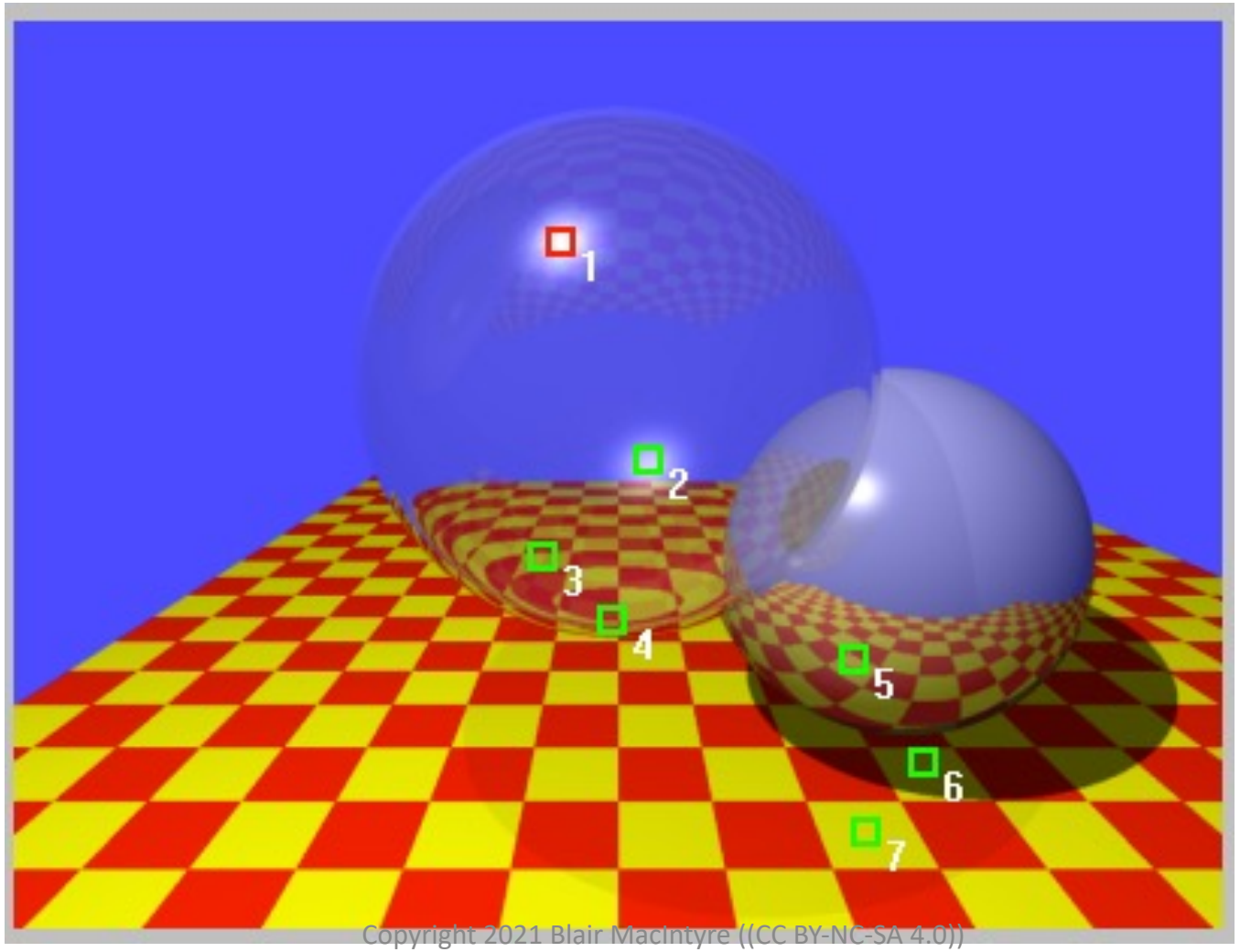
Basic Idea



Basic Algorithm

from *Alan Watt, 3D Computer Graphics, 3rd Edition, 1999*

The Adventures of 7 Rays

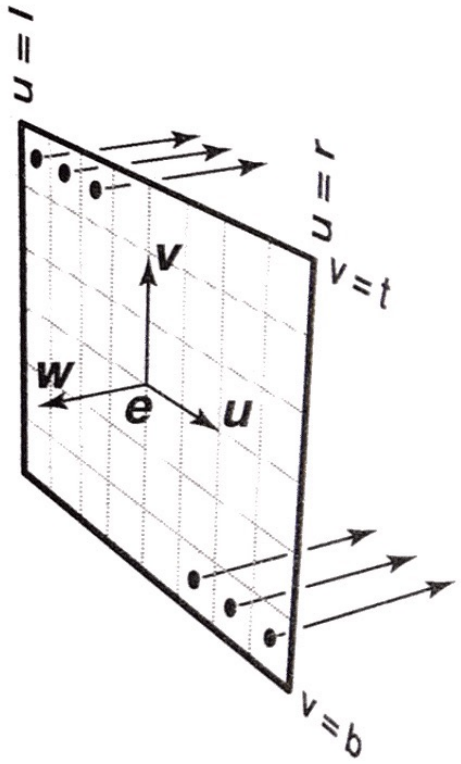


Basic Algorithm

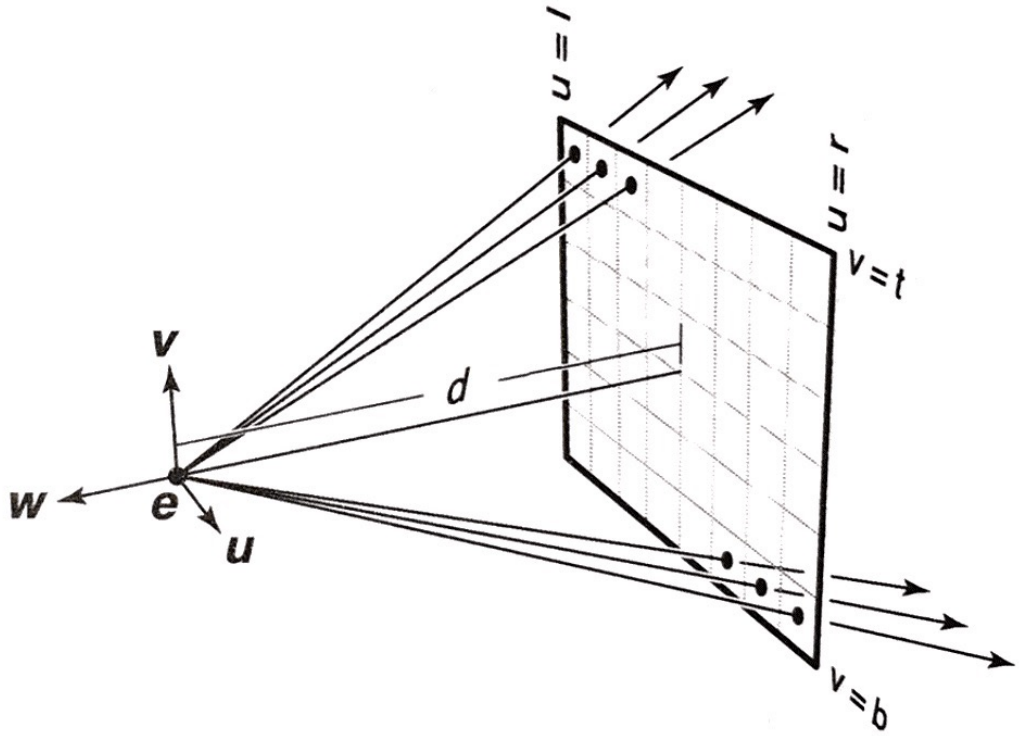
Illumination of a point

$$L = k_a I_a + k_s L_r + k_s L_r + \sum_{1 \le i \le N} S_i I_i \left[k_d (N \cdot L_i) + k_s (R_i \cdot V)^{p_i} \right]$$

Eye Rays: Depends on Projection (Orthographic, Perspective, Oblique)

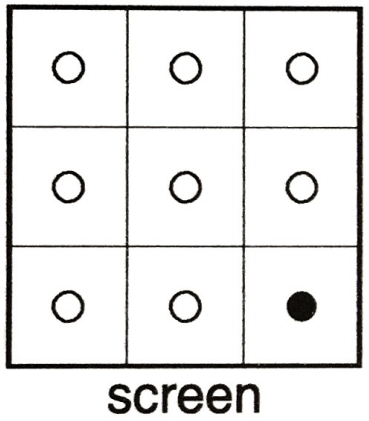


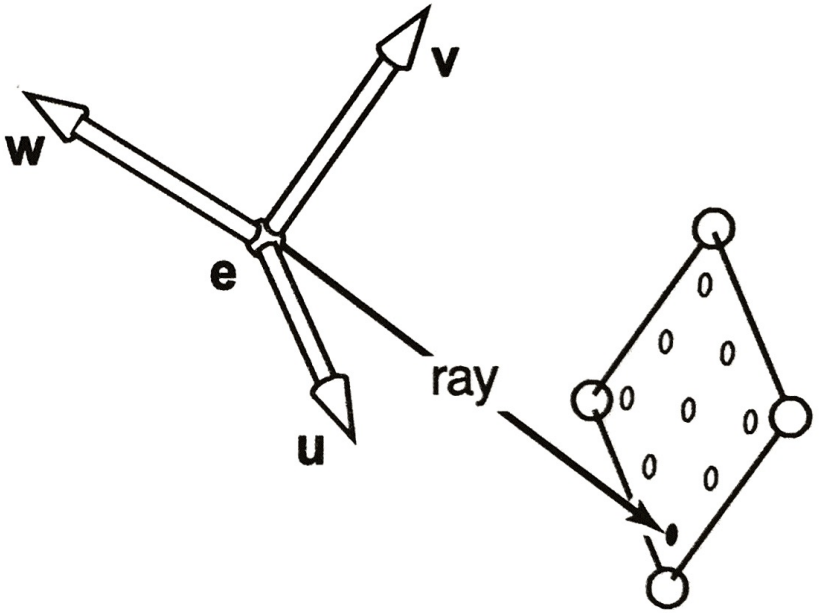
Parallel projection
same direction, different origins

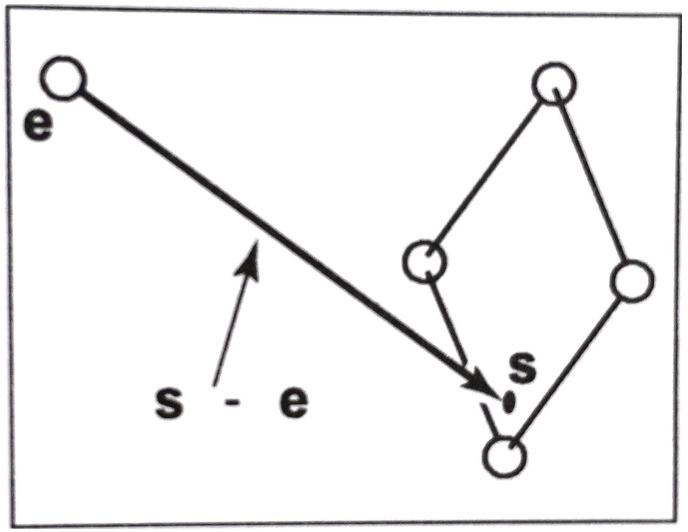


Perspective projection
same origin, different directions

$$p(t) = e + t(s - e)$$





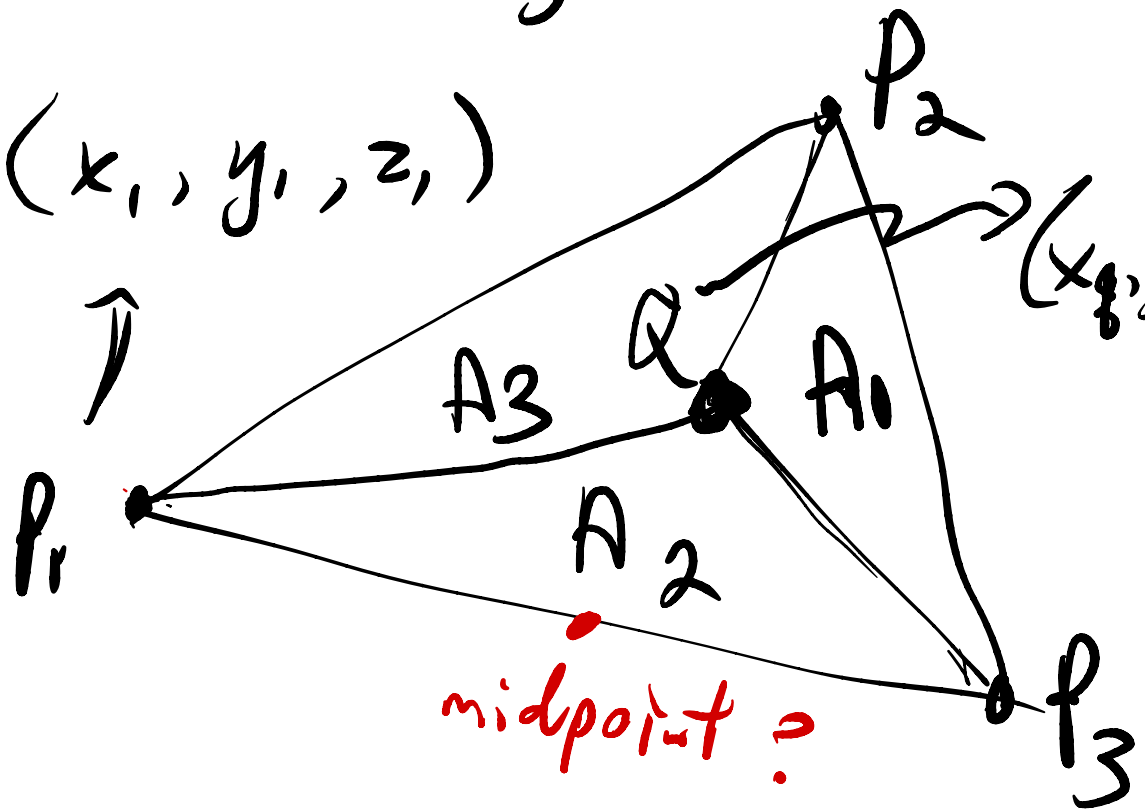


Computing Intersections

Sphere/Ray Intersections

Ray/Triangle Intersection

Barycentric Coordinates



A_1 = area of sub-triangle opposite P_1 (A_2, A_3) ...

$$A = A_1 + A_2 + A_3$$

$$\alpha = A_1 / A$$

$$\beta = A_2 / A$$

$$\gamma = A_3 / A$$

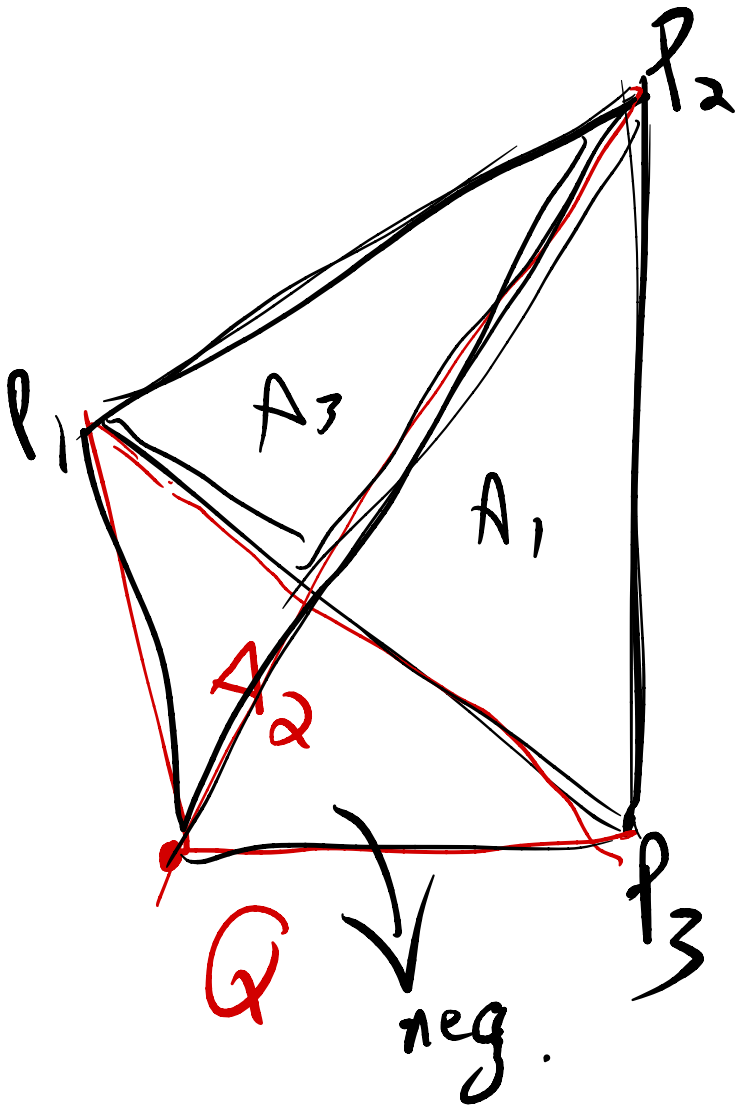
$$\alpha + \beta + \gamma = 1$$

$$Q = \alpha P_1 + \beta P_2 + \gamma P_3$$

$$Q = \alpha P_1 + \beta B_2 + \gamma P_3$$

α, β, γ are positive inside tri

one or more negative if point is outside



Computing Plane Intersection: Implicit Line Equation