Ray tracing

For each pixel shoot a ray R **from camera;**

```
    pixel = TraceRay(R)
```

The recursive ray tracing procedure:

`RGBvalue TraceRay(Ray R)`

- **shoot rays to all light sources;**
- **for all visible sources, compute RGB values r_i**
- **shoot reflected ray R_{refl}; r_{refl}=TraceRay(R_{refl})**
- **shoot refracted ray R_{trans};r_{trans}= TraceRay(R_{trans})**
- **compute resulting RGB value from r_i, r_{refl}, r_{trans} using the lighting model**

Some primitives

Finite primitives:

- polygons
- spheres, cylinders, cones
- parts of general quadrics

Infinite primitives:

- planes
- infinite cylinders and cones
- general quadrics

A finite primitive is often an intersection of an infinite with an area of space

Intersecting rays with objects

General approach:

Use whenever possible the implicit equation $F(q) = 0$ of the object or object parts. Use parametric equation of the line of the ray, $q = p+vt$.

Solve the equation $F(p+vt) = 0$ to find possible values of t. Find the minimal nonnegative value of t to get the intersection point (checking that t is nonegative is important: we want intersections with the ray starting from p, not with the whole line!

General quadrics

A general quadric has equation

$Ax^2 + By^2 + Cz^2 + Dxy + Eyz + Fxz + Gx + Hy + Iz + J = 0$

Intersections with general quadrics are computed in a way similar to cones and cylinders: for a ray p+ v t, take $x = p^x + v^x t$, $y = p^y + v^y t$, $y = p^z + v^z t$,

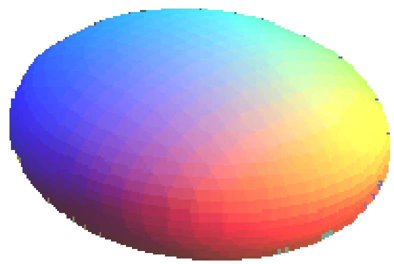
and solve the equation for t; if there are solutions,

take the smaller nonnegative one.

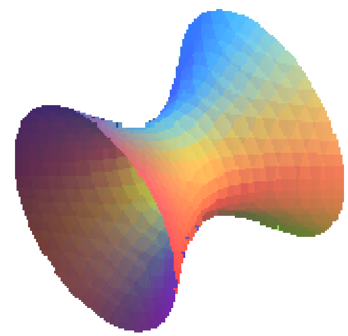
Infinite cones and cylinders are special cases of general quadrics.

General quadrics

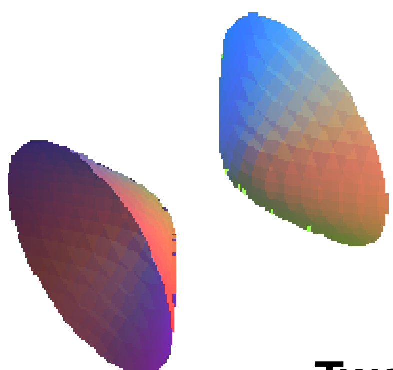
Nondegenerate quadrics



Ellipsoid
$x^2/a^2 + y^2/b^2 + z^2/c^2 + 1 = 0$



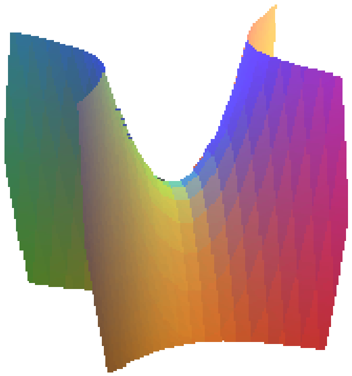
One-sheet hyperboloid
$x^2/a^2 - y^2/b^2 + z^2/c^2 + 1 = 0$



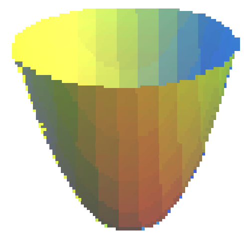
Two-sheet hyperboloid
$x^2/a^2 - y^2/b^2 + z^2/c^2 - 1 = 0$

General quadrics

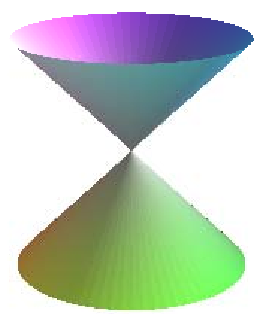
Nondegenerate quadrics



Hyperbolic parabolid
$x^2/a^2 - z^2/c^2 - 2y = 0$



Elliptic parabolid
$x^2/a^2 + z^2/c^2 - 2y = 0$



cone
$x^2/a^2 - y^2/b^2 + z^2/c^2 = 0$

General quadrics

Degenerate quadrics

- **planes (no quadratic terms),**
- **pairs of parallel planes (e.g. $x^2 - 1 = 0$)**
- **pairs of intersecting planes (e.g. $x^2-1 = 0$)**
- **elliptic cylinders (e.g. $x^2+z^2-1=0$)**
- **hyperbolic cylinders (e.g. $x^2-z^2-1=0$)**
- **parabolic cylinders (e.g. $x^2 -z = 0$)**

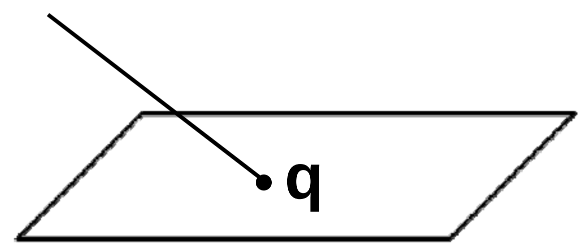
Possible to get “imaginary” surfaces (that is, with no points)! Example: $x^2 + 1 = 0$

Polygon-ray intersections

Two steps:

- intersect with the plane of the polygon
- check if the intersection point is inside the polygon

We know how to compute intersections with the plane (see prev. lecture). Let $q=[q_x,q_y,q_z]$ be the intersection point.



Polygon-ray intersections

Possible to do the whole calculation using 3d points, but it is more efficient to use 2d points.

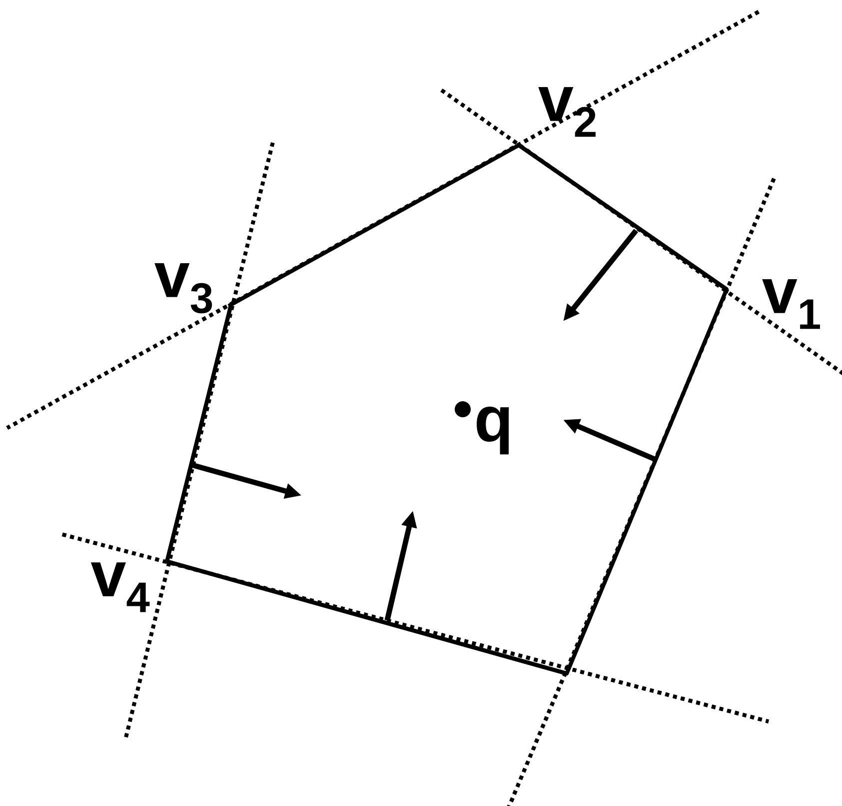
Converting to the coordinates in the plane is computationally expensive. Idea: project to a coordinate plane (XY, YZ, or XZ) by discarding one of the vector coordinates.

We cannot always discard, say, Z, because the polygon may project to an interval, if it is in a plane parallel to Z.

Choose the coordinate to discard so that the corresponding component of the normal to the polygon is maximal. E.g. if $n_x > n_y$ and $n_x > n_z$, discard X.

2D Polygon-ray intersections

Now we can assume that all vertices v_i and the intersection point q are 2D points.



Assume that polygons are convex. A convex polygon is the intersection of a set of half-planes, bounded by the lines along the polygon edges. To be inside the polygon the point has to be in each half-plane. Recall that the implicit line equation can be used to check on which side of the line a point is.

2D Polygon-ray intersections

Equation of the line through the edge connecting vertices $\mathbf{v_i}$ and $\mathbf{v_{i+1}}$:

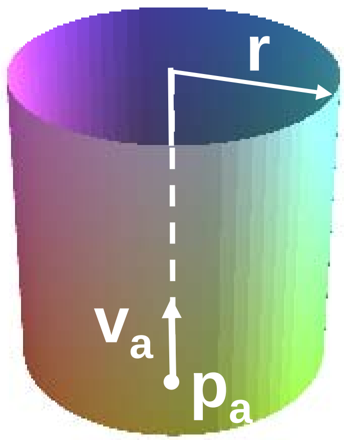
$$\left(\mathbf{v}_{\mathbf{i}}^{\mathbf{y}} - \mathbf{v}_{\mathbf{i}+1}^{\mathbf{y}}\right)\left(\mathbf{x} - \mathbf{v}_{\mathbf{i}}^{\mathbf{x}}\right) + \left(\mathbf{v}_{\mathbf{i}+1}^{\mathbf{x}} - \mathbf{v}_{\mathbf{i}}^{\mathbf{x}}\right)\left(\mathbf{y} - \mathbf{v}_{\mathbf{i}}^{\mathbf{y}}\right) = 0$$

If the quantity on the right-hand side is positive, then the point (x,y) is to the right of the edge, assuming we are looking from $\mathbf{v_i}$ to $\mathbf{v_{i+1}}$.

Algorithm: if for each edge the quantity above is nonnegative for $\mathbf{x = q^x}$, $\mathbf{y = q^y}$ then the point q is in the polygon. Otherwise, it is not.

In the formulas x and y should be replaced by x and z if y coord. was dropped, or by y and z if x was dropped.

Infinite cylinder-ray intersections



Infinite cylinder along y of radius r axis has equation $x^2 + z^2 - r^2 = 0$.
The equation for a more general cylinder of radius r oriented along a line $p_a + v_a t$:
$(q - p_a - (v_a \cdot (q - p_a))\, v_a)^2 - r^2 = 0$
where q = (x,y,z) is a point on the cylinder.

Infinite cylinder-ray intersections

To find intersection points with a ray p + vt, substitute q = p + vt and solve:

$$(\mathbf{p} - \mathbf{p}_a + \mathbf{v}t - (\mathbf{v}_a \cdot (\mathbf{p} - \mathbf{p}_a + \mathbf{v}t))\mathbf{v}_a)^2 - r^2 = 0$$

reduces to $\mathbf{A}t^2 + \mathbf{B}t + \mathbf{C} = 0$

with

$$\mathbf{A} = \left(\mathbf{v} - (\mathbf{v} \cdot \mathbf{v}_a)\mathbf{v}_a\right)^2$$

$$\mathbf{B} = 2\left((\mathbf{v} - (\mathbf{v} \cdot \mathbf{v}_a)\mathbf{v}_a) \cdot (\Delta\mathbf{p} - (\Delta\mathbf{p} \cdot \mathbf{v}_a)\mathbf{v}_a)\right)$$

$$\mathbf{C} = \left(\Delta\mathbf{p} - (\Delta\mathbf{p} \cdot \mathbf{v}_a)\mathbf{v}_a\right)^2 - r^2$$

where $\Delta\mathbf{p} = \mathbf{p} - \mathbf{p}_a$

Cylinder caps

A finite cylinder with caps can be constructed as the intersection of an infinite cylinder with a slab between two parallel planes, which are perpendicular to the axis.

To intersect a ray with a cylinder with caps:

- intersect with the infinite cylidner;
- check if the intersection is between the planes;
- intersect with each plane;
- determine if the intersections are inside caps;
- out of all intersections choose the on with minimal t

Cylinder-ray intersections

POV -ray like cylinder with caps : cap centers at p_1 and p_2, radius r.

Infinite cylinder equation: $p_a = p_1$, $v_a = (p_2 - p_1)/|p_2 - p_1|$

The finite cylinder (without caps) is described by equations:

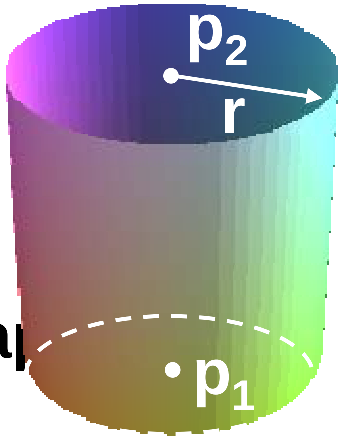
$(q - p_a - (v_a \cdot (q - p_a))v_a)^2 - r^2 = 0$ and $(v_a \cdot (q - p_1)) > 0$ and

$(v_a \cdot (q - p_2)) < 0$

The equations for caps are:

$(v_a \cdot (q - p_1)) = 0, (q - p_1)^2 < r^2$ bottom cap

$(v_a \cdot (q - p_2)) = 0, (q - p_2)^2 < r^2$ top cap



Cylinder-ray intersections

Algorithm with equations:

Step 1: Find solutions t_1 and t_2 of $At^2 + Bt + C = 0$

if they exist. Mark as intersection candidates the one(s) that are nonnegative and for which $(v_a \cdot (q_i - p_1)) > 0$ and $(v_a \cdot (q_i - p_2)) < 0$, where $q_i = p + v\, t_i$

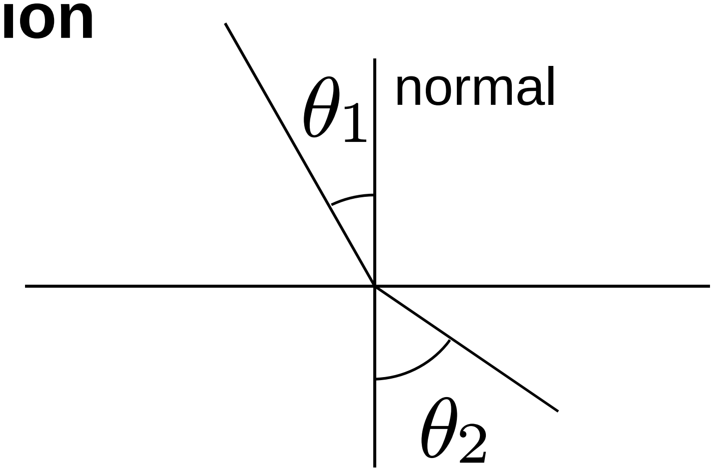
Step 2: Compute t_3 and t_4, the parameter values for which the ray intersects the upper and lower planes of the caps.
If these intersections exists, mark as intersection candidates those that are nonegative and $(q_3 - p_1)^2 < r^2$ (respectively $(q_4 - p_2)^2 < r^2$).

In the set of candidates, pick the one with min. t.

Snell's law

If a surface separates two media with different refraction indices (e.g. air and water) the light rays change direction when they go through.

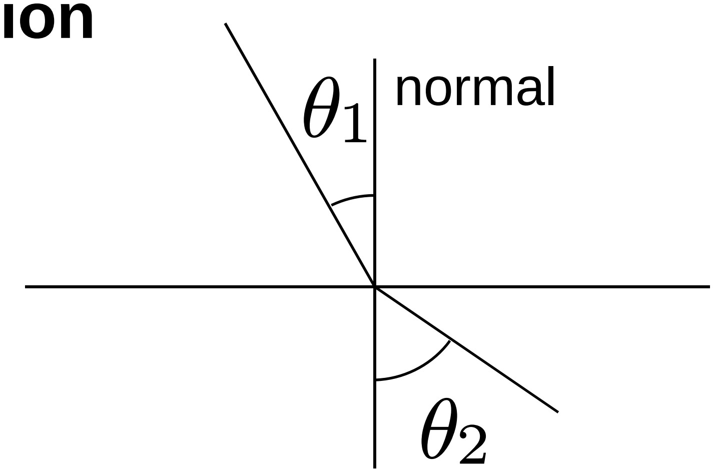
<u>Snell's law</u>: the refracted ray is stays in the plane spanned by the normal and the direction of the original ray. The angles between the normal and the rays are related by $n_1 \sin\theta_1 = n_2 \sin\theta_2$ **where** n_1 **and** n_2 **are refraction indices.**



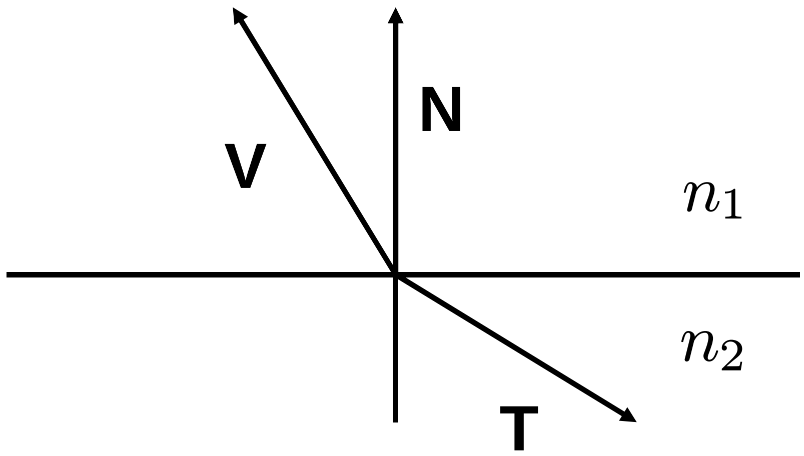
Snell's law

If a surface separates two media with different refraction indices (e.g. air and water) the light rays change direction when they go through.

<u>Snell's law</u>: the refracted ray is stays in the plane spanned by the normal and the direction of the original ray. The angles between the normal and the rays are related by $n_1 \sin\theta_1 = n_2 \sin\theta_2$ **where** n_1 **and** n_2 **are refraction indices.**



Refracted ray direction



$$n = \frac{n_1}{n_2}$$

$$\mathbf{T} = -n\mathbf{V} + \left(n(\mathbf{V} \cdot \mathbf{N}) - \sqrt{1 - n^2(1 - (\mathbf{V} \cdot \mathbf{N})^2)} \right) \mathbf{N}$$