Simple Intersection

jodavis42@gmail.com

Simple shape review

Primitives:

- Point
- Plane
- Triangle
- Aabb
- Sphere
- Ray
- Frustum

Plane

Point + Normal

$$\vec{n} \cdot (\vec{p} - \vec{p}_0) = 0$$

Requires 6 floats

Expand to $\vec{n} \cdot \vec{p} = d$

```
struct Plane
{
  // (n.x, n.y, n.z, d)
  Vector4 mData;
};
```

Triangle

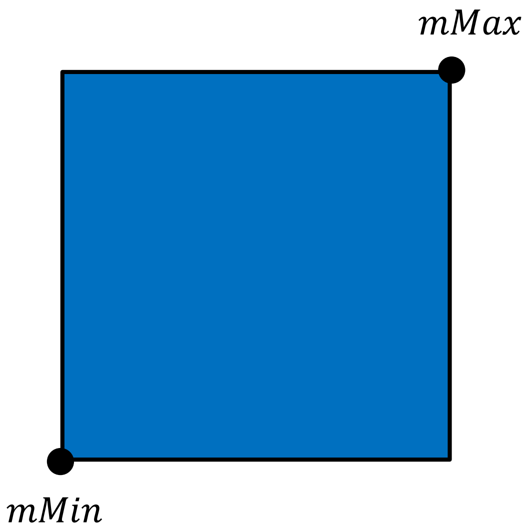
Nothing special

```
struct Triangle
{
  Vector3 mP0;
  Vector3 mP1;
  Vector3 mP2;
};
```

Axis Aligned Bounding Box (Aabb)

Min and max on each axis

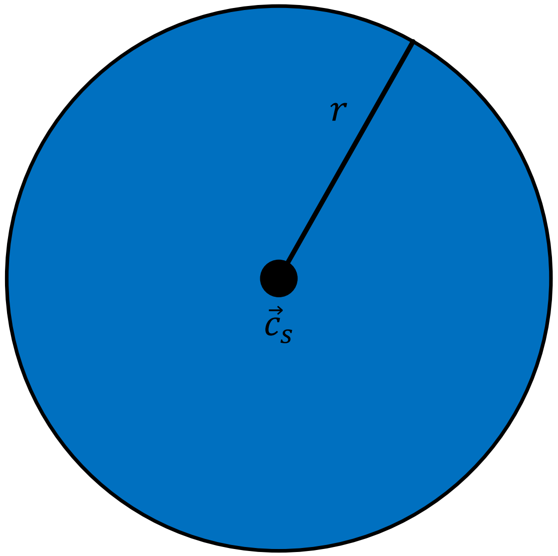
```
struct Aabb
{
  Vector3 mMin;
  Vector3 mMax;
};
```



Sphere

Sphere equation: $(\vec{c}_s - \vec{p})^2 - r^2 = 0$

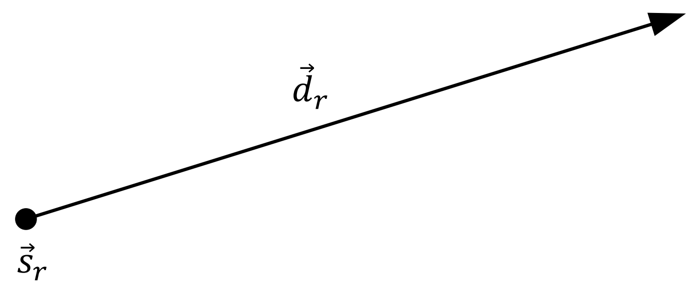
```
struct Sphere
{
  Vector3 mPosition;
  float mRadius;
};
```



Ray

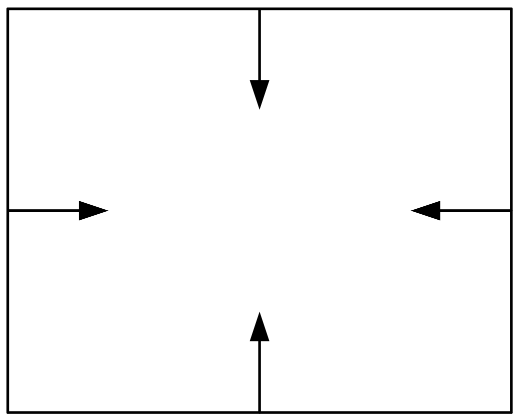
Ray equation: $\vec{p}_r(t) = \vec{s}_r + \vec{d}_r t$

```
struct Ray
{
  Vector3 mStart;
  Vector3 mDirection;
};
```



Frustum

```
struct Frustum
{
  Plane mPlanes[6];
  Vector3 mPoints[8];
};
```

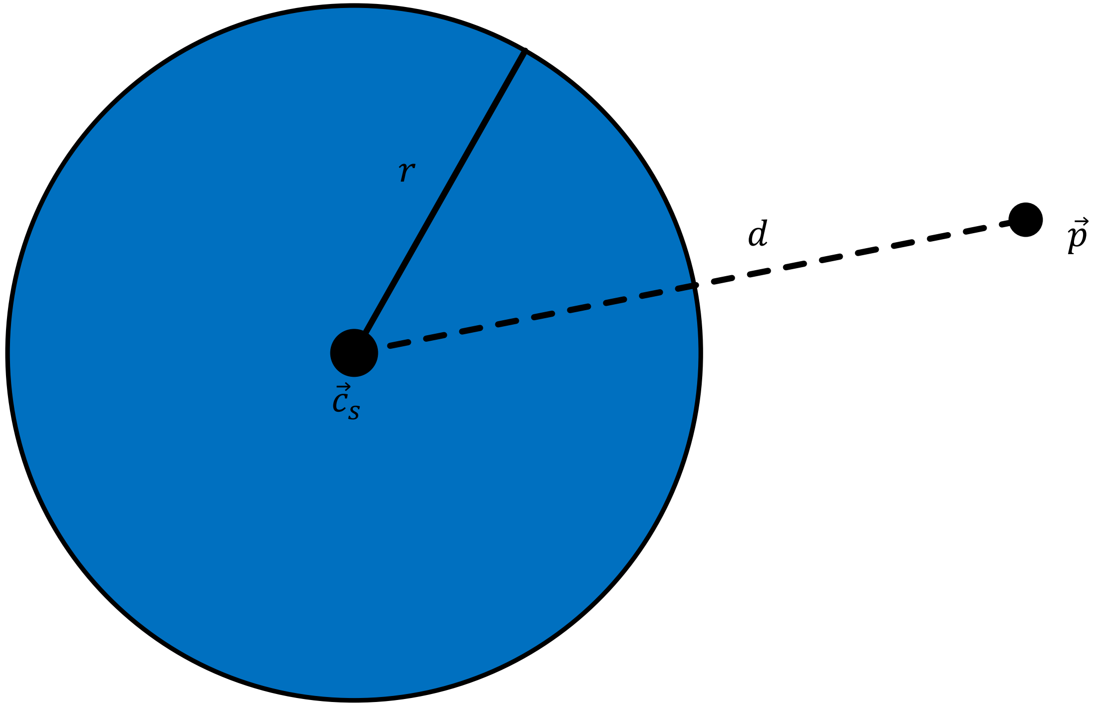


Normals point inwards

Intersection Test Types

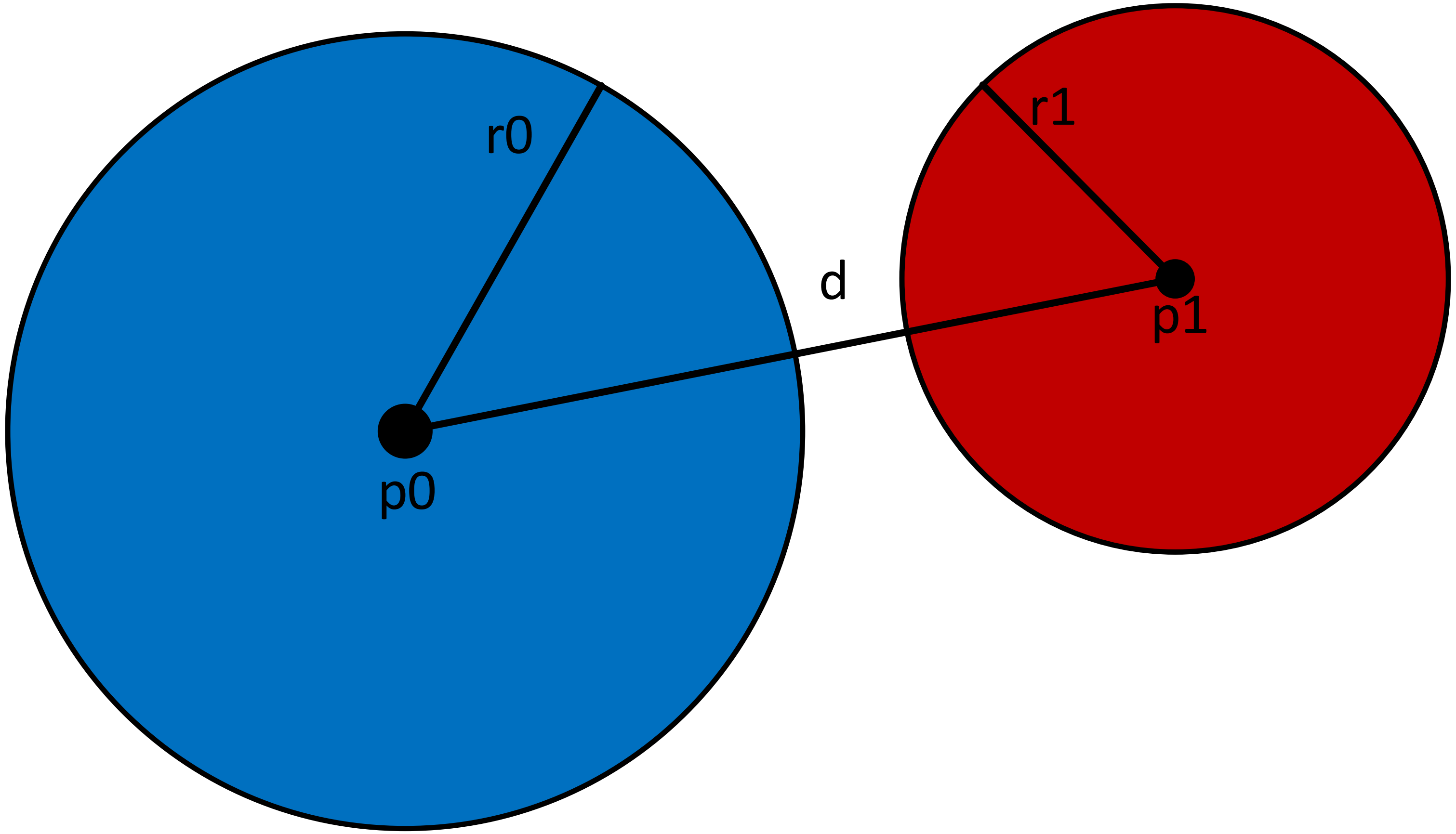
- Boolean
- Containment
  - Coplanar, Outside, Inside, Overlap
- Intersection
  - Same as containment but typically with a t-value

Point vs. Sphere

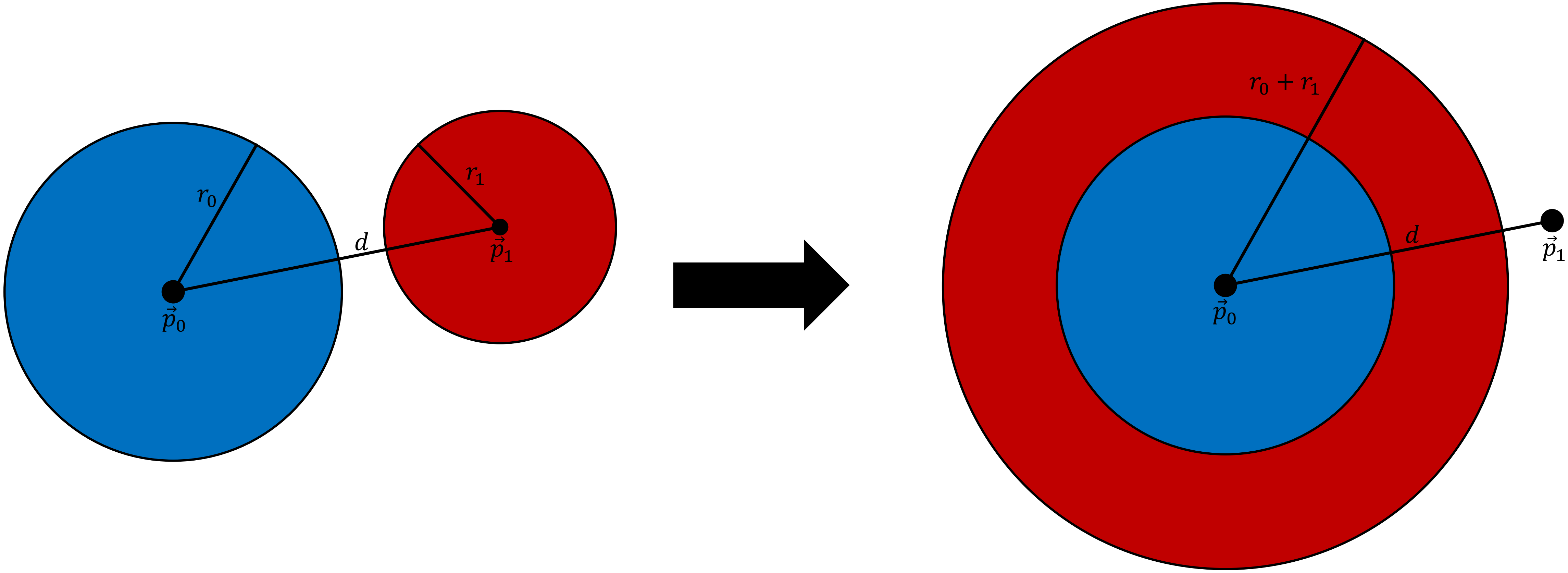


If $d \leq r$ then the point is contained

Sphere vs. Sphere



Sphere vs. Sphere (Alternate)

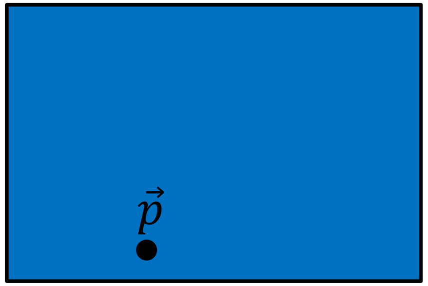


Conceptually expand one sphere by the other's radius
then test point for containment

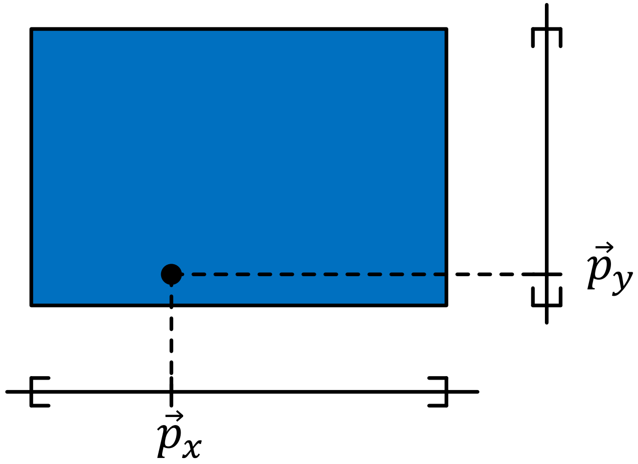
Point vs. Aabb

Each axis of an aabb is independent.

Instead of Point vs. Aabb



Test each axis independently



Defining a test for one dimension is easy.

We can extend to n dimensions later.

Point vs. Aabb – 1 Dimension Test

How do we test one axis?

Two main ways:

Intersection Test: $min \leq p \leq max$

Non-Intersection Test: $p < min \text{ or } p > max$

Point vs. Aabb – n Dimensions

How do we combine the 1-dimensions test to get n-dimensions?

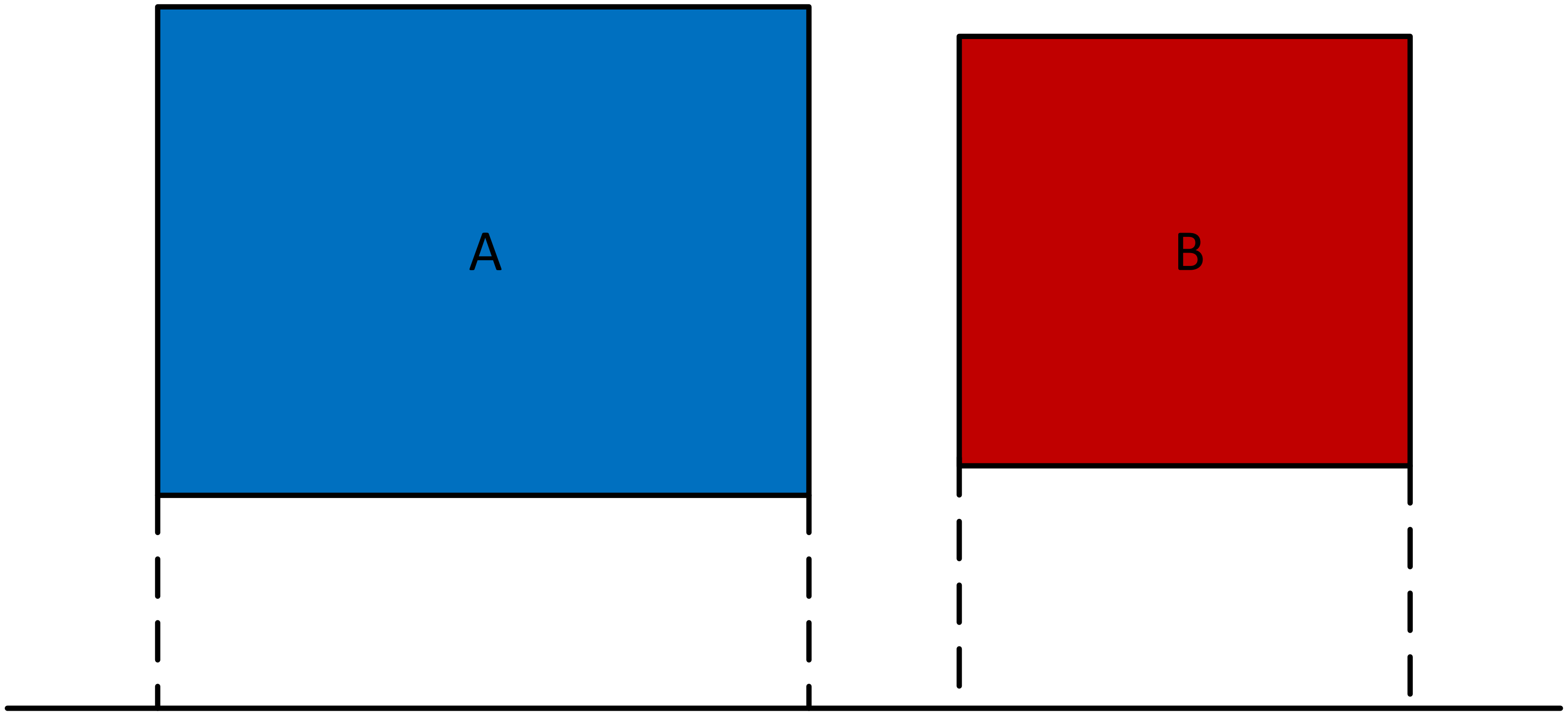
Intersection Test: If all axes are contained

Non-Intersection Test: If any axis isn't contained

*Some tests will be much easier to write for non-intersection

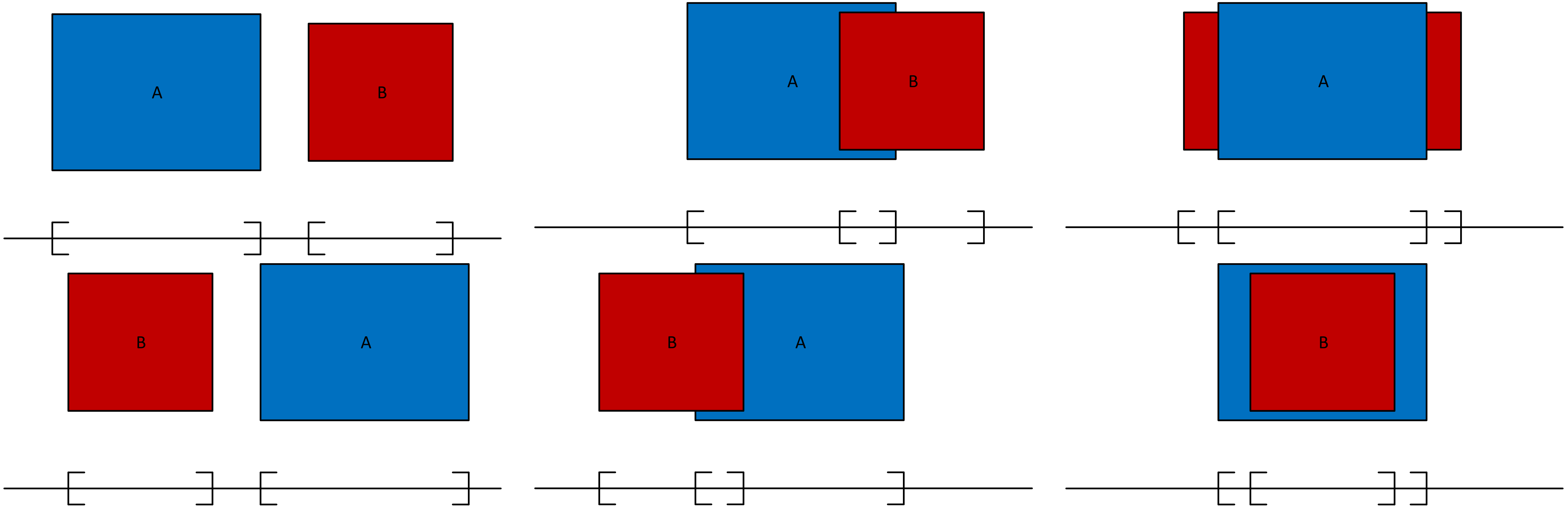
Aabb vs. Aabb

First look at one axis.



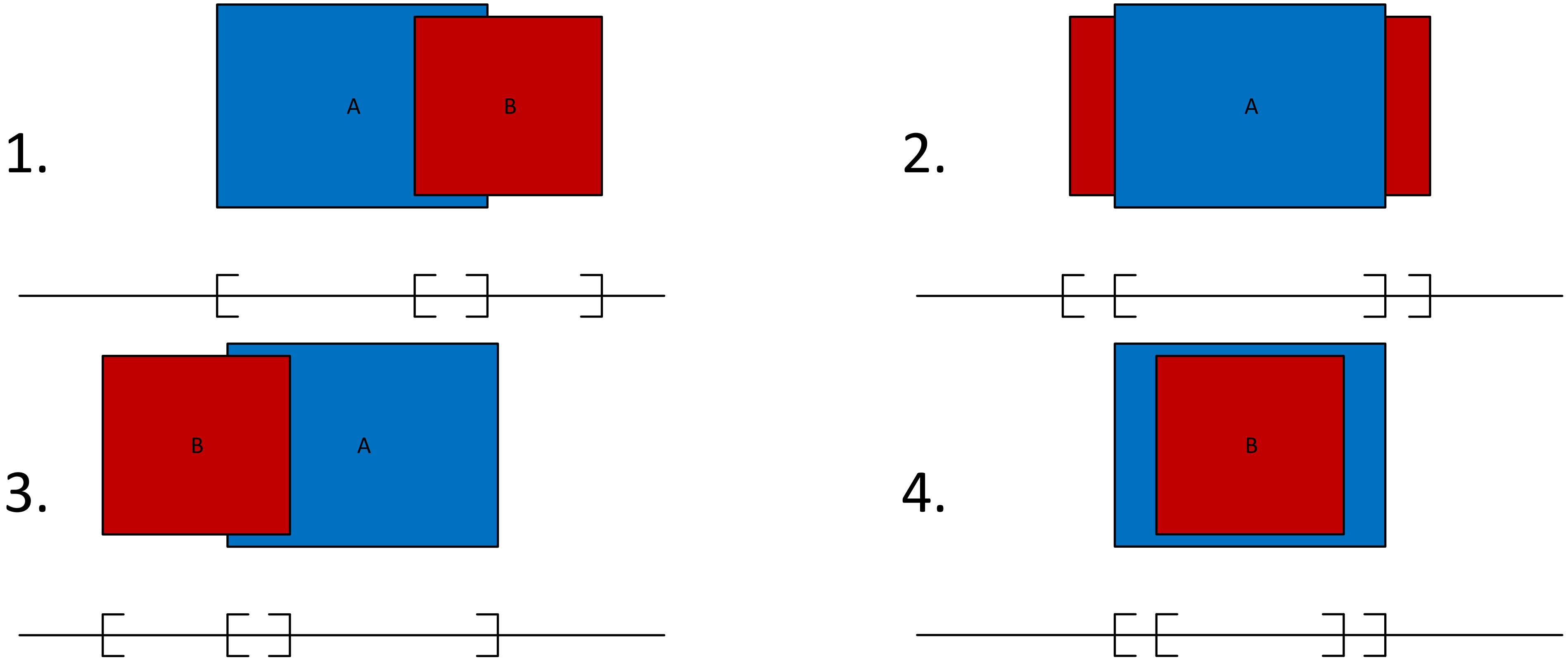
Aabb vs. Aabb

6 Cases to consider



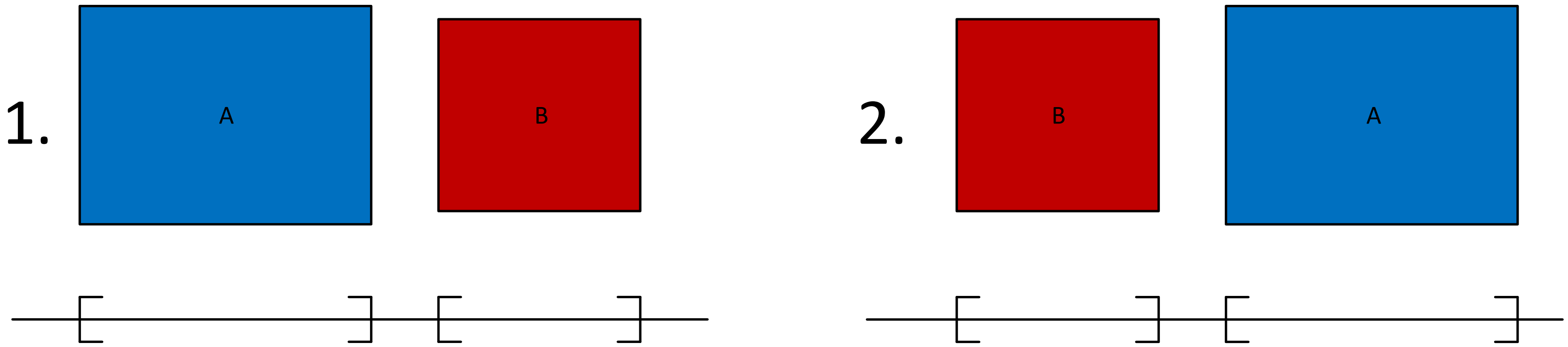
Aabb vs. Aabb

How can we write a test for intersection from these 4 cases?



Aabb vs. Aabb

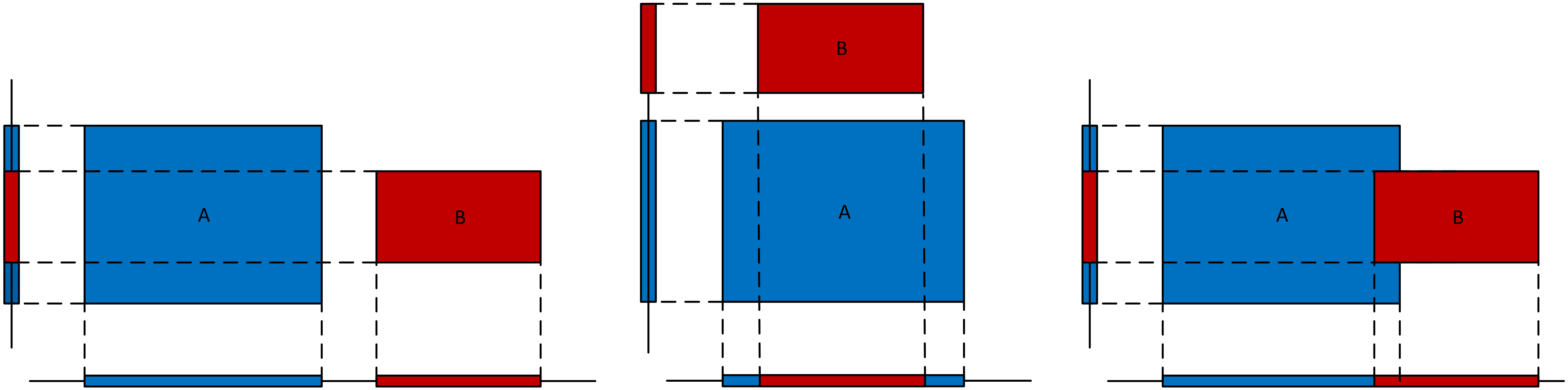
How can we write a test for non-intersection from these 2 cases?



Aabb vs. Aabb

How do we combine tests for non-intersection?

If an axis is separating then there's no intersection



Ray vs. Plane

Given:

Ray: $\vec{p}_r(t) = \vec{s}_r + \vec{d}_r t$

Plane: $\vec{n} \cdot (\vec{p} - \vec{p}_0) = 0$

How do we solve?

What are we solving for?

Ray vs. Plane

We had a third equation we forgot about.

Given:

$$\vec{p}_r(t) = \vec{s}_r + \vec{d}_r$$
$$\vec{n} \cdot (\vec{p} - \vec{p}_0) = 0$$
$$\vec{p} = \vec{p}_r(t)$$

Substitute:

$$\vec{n} \cdot (\vec{p} - \vec{p}_0) = 0$$
$$\vec{n} \cdot (\vec{p}_r(t) - \vec{p}_0) = 0$$
$$\vec{n} \cdot \left(\vec{s}_r + \vec{d}_r t - \vec{p}_0\right) = 0$$

Solve for t

Ray vs. Plane

What do we have to consider before finishing?

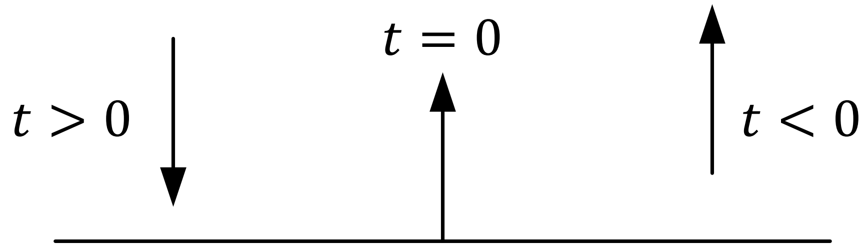
1. When can this fail to give a t-value?

$$t = \frac{\vec{n} \cdot (\vec{p}_0 - \vec{s}_r)}{\vec{n} \cdot \vec{d}_r}$$

Ray vs. Plane

What do we have to consider before finishing?

2. Are all values of t valid?



Ray vs. Triangle

A triangle defines a plane

We know how to test Ray vs. Plane

If we can define Point vs. Triangle we know Ray vs. Triangle

How do we test Point vs. Triangle?

Barycentric Coordinates - Triangle

Barycentric coordinates are defined as:

$$\vec{P} = u\vec{A} + v\vec{B} + w\vec{C}$$
$$u + v + w = 1$$

1 coordinate is redundant:

$$w = 1 - u - v$$

If $0 \leq u, v, w \leq 1$ then $\vec{P}$ is inside the triangle

How do we compute u and v?

Barycentric Coordinates - Triangle

How do we solve $\vec{P} = u\vec{A} + v\vec{B} + w\vec{C}$?

We have 3 equations and 3 unknowns!

$$\begin{bmatrix} P_x \\ P_y \\ P_z \end{bmatrix} = u \begin{bmatrix} A_x \\ A_y \\ A_z \end{bmatrix} + v \begin{bmatrix} B_x \\ B_y \\ B_z \end{bmatrix} + w \begin{bmatrix} C_x \\ C_y \\ C_z \end{bmatrix}$$

Barycentric Coordinates - Triangle

Re-arrange to make life easier:

$$\begin{bmatrix} P_x \\ P_y \\ P_z \end{bmatrix} = \begin{bmatrix} A_x & B_x & C_x \\ A_y & B_y & C_y \\ A_z & B_z & C_z \end{bmatrix} \begin{bmatrix} u \\ v \\ w \end{bmatrix}$$

And now we can simply invert:

$$\begin{bmatrix} A_x & B_x & C_x \\ A_y & B_y & C_y \\ A_z & B_z & C_z \end{bmatrix}^{-1} \begin{bmatrix} P_x \\ P_y \\ P_z \end{bmatrix} = \begin{bmatrix} u \\ v \\ w \end{bmatrix}$$

Any issues?

When is a matrix inverse not defined?

Barycentric Coordinates - Triangle

A matrix $\boldsymbol{M}$ is invertible if and only if its determinant is non-zero.

Is this matrix's determinant always non-zero? $\begin{bmatrix} A_x & B_x & C_x \\ A_y & B_y & C_y \\ A_z & B_z & C_z \end{bmatrix}^{-1}$

Barycentric coordinates - Triangle

Scalar Triple Product:

$$det\begin{pmatrix} A_x & B_x & C_x \\ A_y & B_y & C_y \\ A_z & B_z & C_z \end{pmatrix} = \vec{A} \cdot \left(\vec{B} \times \vec{C}\right)$$

When is this zero?

What is wrong with this formula?

Barycentric coordinates - Triangle

We thought we had 3 equations and 3 unknowns…

We actually have 4 equations and 3 unknowns

$$\vec{P} = u\vec{A} + v\vec{B} + w\vec{C}$$
$$u + v + w = 1$$

How do we solve now?

Barycentric coordinates - Triangle

Knowing:

$$w = 1 - u - v$$

$$\vec{P} = u\vec{A} + v\vec{B} + (1 - u - v)\vec{C}$$

Then re-arrange: $\vec{P} - \vec{C} = u(\vec{A} - \vec{C}) + v(\vec{B} - \vec{C})$

Now we have 3 equations and 2 unknowns…

Barycentric coordinates - Triangle

First define:

$$\vec{v}_0 = \vec{P} - \vec{C}$$
$$\vec{v}_1 = \vec{A} - \vec{C}$$
$$\vec{v}_2 = \vec{B} - \vec{C}$$

Now we have: $\vec{v}_0 = u\vec{v}_1 + v\vec{v}_2$

Can turn this into 2 equations by projecting on $\vec{v}_1$ and $\vec{v}_2$

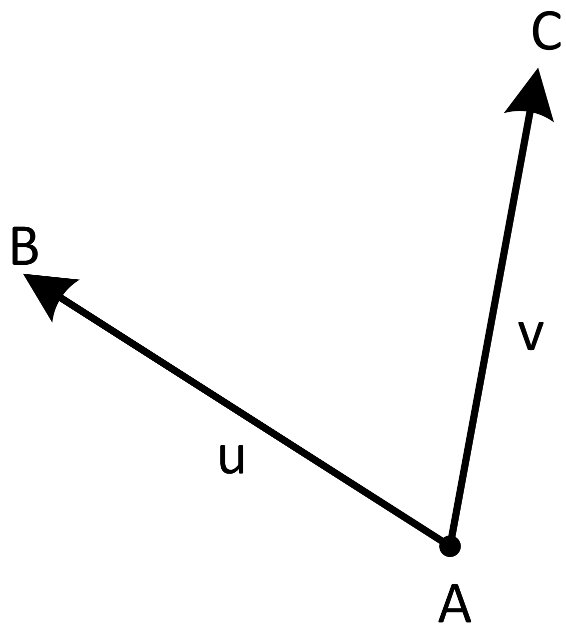
$$\vec{v}_0 \cdot \vec{v}_1 = u(\vec{v}_1 \cdot \vec{v}_1) + v(\vec{v}_2 \cdot \vec{v}_1)$$
$$\vec{v}_0 \cdot \vec{v}_2 = u(\vec{v}_1 \cdot \vec{v}_2) + v(\vec{v}_2 \cdot \vec{v}_2)$$

Cramer's Rule

$$ax + by = e$$
$$cx + dy = f$$

$$x = \frac{\begin{vmatrix} e & b \\ f & d \end{vmatrix}}{\begin{vmatrix} a & b \\ c & d \end{vmatrix}} \qquad y = \frac{\begin{vmatrix} a & e \\ c & f \end{vmatrix}}{\begin{vmatrix} a & b \\ c & d \end{vmatrix}}$$

Barycentric coordinates - Triangle

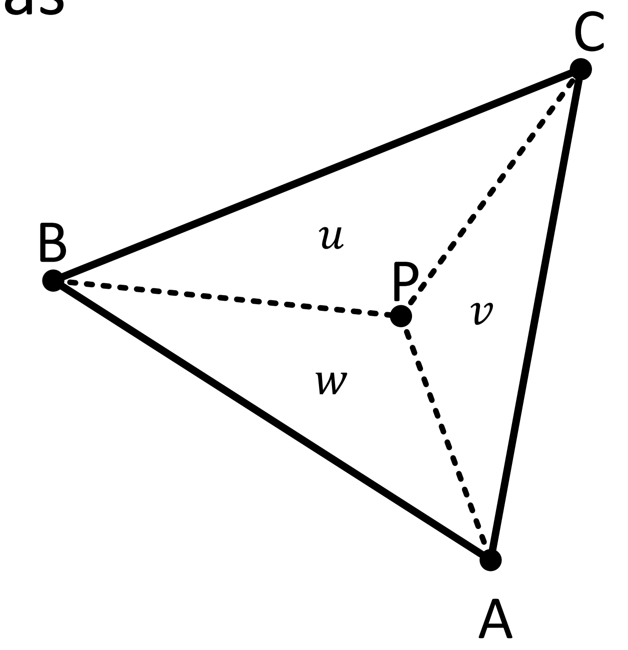


u and v are a ratios on the edges $(\vec{B} - \vec{A})$ and $(\vec{C} - \vec{A})$

Barycentric coordinates (areal coordinates)

Method 3: Signed triangle area ratio

Coordinates are proportional to signed ratio areas



How do we get the area of a triangle?

Barycentric coordinates (areal coordinates)

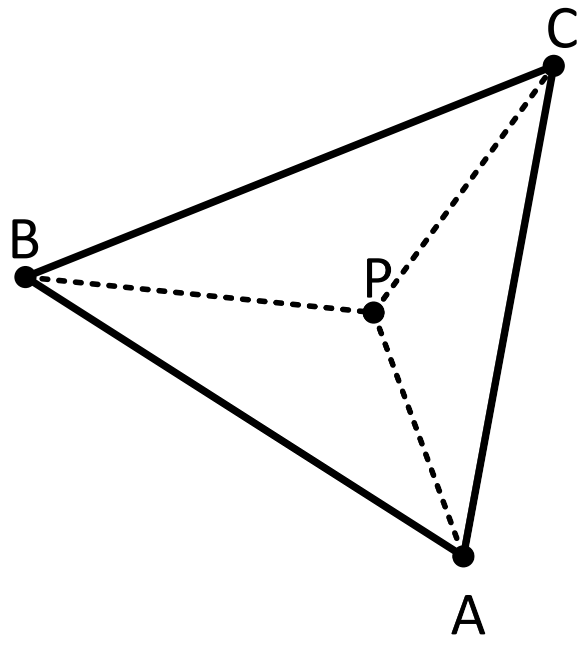
Cross product defines the area of a parallelogram

Area of a triangle is: $A = \frac{1}{2}\left|\left(\vec{B} - \vec{A}\right) \times \left(\vec{C} - \vec{A}\right)\right|$

What about the signed area?

Barycentric coordinates (areal coordinates)

The sub-triangle PBC defines the normal $\vec{N}_{PBC}$



Now we can define signed area:

$$SA = \frac{1}{2}\vec{N}_{PBC} \cdot \frac{\vec{N}_{ABC}}{|\vec{N}_{ABC}|}$$

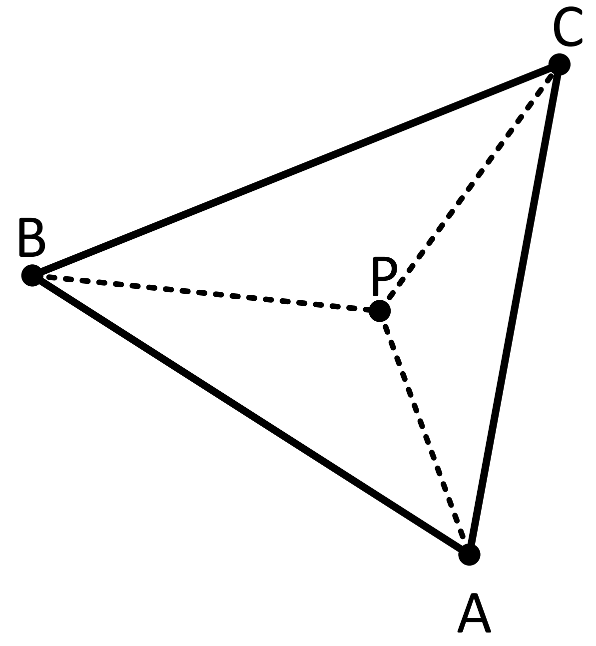
If the winding order flips, so does the sign

Barycentric coordinates (areal coordinates)

Barycentric coordinates are a ratio of singed areas

$$u = \frac{SA(PBC)}{SA(ABC)}$$

$$v = \frac{SA(PCA)}{SA(ABC)}$$



These can be simplified: $u = \dfrac{\vec{N}_{PBC} \cdot \vec{N}_{ABC}}{\vec{N}_{ABC} \cdot \vec{N}_{ABC}}$

Barycentric coordinates - Line

$$\vec{P} = u\vec{A} + v\vec{B}$$
$$u + v = 1$$

How do we compute the barycentric coordinates of a line?

2 main approaches like before:

- Analytic
- Geometric

Barycentric coordinates – Line (Analytic)

Solve like before:

$$\vec{P} = u\vec{A} + v\vec{B}$$
$$\vec{P} = u\vec{A} + (1-u)\vec{B}$$
$$\vec{P} - \vec{B} = u\left(\vec{A} - \vec{B}\right)$$

Multiply both sides by $\left(\vec{A} - \vec{B}\right)$:

$$\left(\vec{P} - \vec{B}\right) \cdot \left(\vec{A} - \vec{B}\right) = u\left(\vec{A} - \vec{B}\right) \cdot \left(\vec{A} - \vec{B}\right)$$
$$\frac{\left(\vec{P} - \vec{B}\right) \cdot \left(\vec{A} - \vec{B}\right)}{\left(\vec{A} - \vec{B}\right) \cdot \left(\vec{A} - \vec{B}\right)} = \mathrm{u}$$

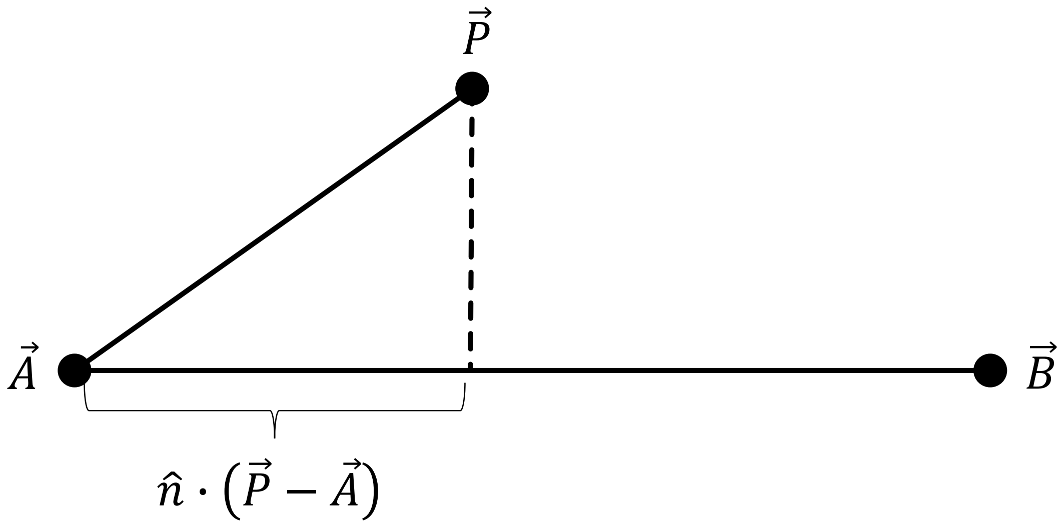
Barycentric coordinates – Line (Geometric)

First let $\vec{n} = \left(\vec{B} - \vec{A}\right)$

Compute $\hat{n} = \dfrac{\vec{n}}{|\vec{n}|}$

Project $\vec{P}$ onto the and solve

$$v = \frac{\hat{n} \cdot \left(\vec{P} - \vec{A}\right)}{\left|\vec{B} - \vec{A}\right|}$$



*Divide by $\left|\vec{B} - \vec{A}\right|$ to “normalize” v

Misc. Barycentric coordinates facts

Can map points between different shapes

Can map points between spaces (including projection)

Can interpolate values (actual triangle rasterization)

Ray vs. Sphere

Given:

Ray: $\vec{p}_r(t) = \vec{s}_r + \vec{d}_r t$

Sphere: $(\vec{c}_s - \vec{p})^2 - r^2 = 0$

$$\vec{p}_r(t) = \vec{p}$$

Substitute:

$$\left(\vec{c}_s - \left(\vec{s}_r + \vec{d}_r t\right)\right)^2 - r^2 = 0$$

We can use the quadratic formula if we re-arrange to:

$$at^2 + bt + c = 0$$

Ray vs. Sphere

Given:

$$\left(\vec{c}_s - \left(\vec{s}_r + \vec{d}_r t\right)\right)^2 - r^2 = 0$$

How do we expand a 3-term square?

$$(a + b + c)^2 = ?$$

Alternatively, we can group knowns together:

$$\left(\vec{m} - \vec{d}_r t\right)^2 - r^2 = 0$$

Ray vs. Sphere – Quadratic Equation

Solve the quadratic equation $at^2 + bt + c = 0$

with the quadratic formula:

$$t = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

What cases do we need to consider?

Ray vs. Sphere – Quadratic Equation

$$t = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

When is the denominator 0?

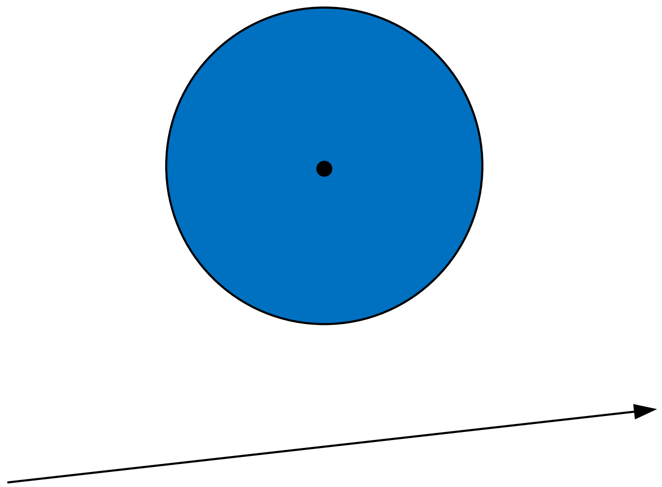
What do the 3 cases of the discriminant (Δ) mean?

$$\Delta < 0$$
$$\Delta > 0$$
$$\Delta = 0$$

Ray vs. Sphere – Quadratic Equation

Case 1: $\Delta < 0$

There is no solution (in Euclidean space)

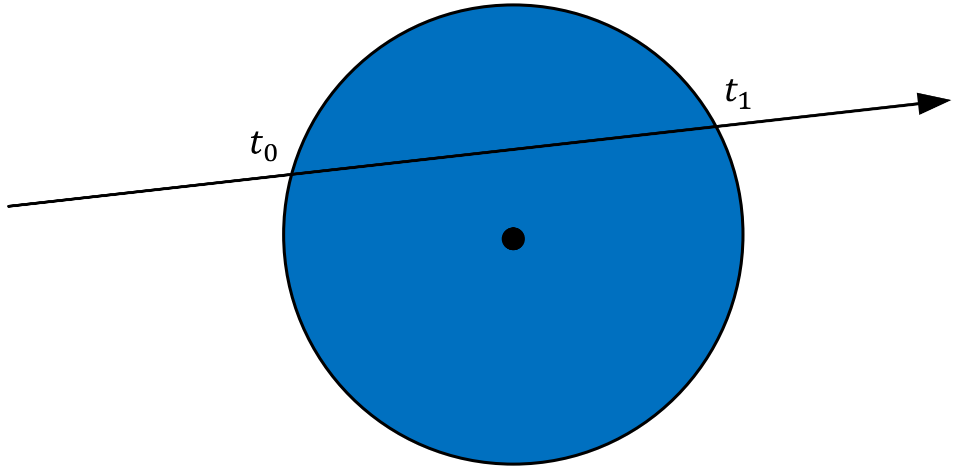


The line doesn't hit the sphere!

Ray vs. Sphere – Quadratic Equation

Case 2: $\Delta > 0$

There are 2 solutions

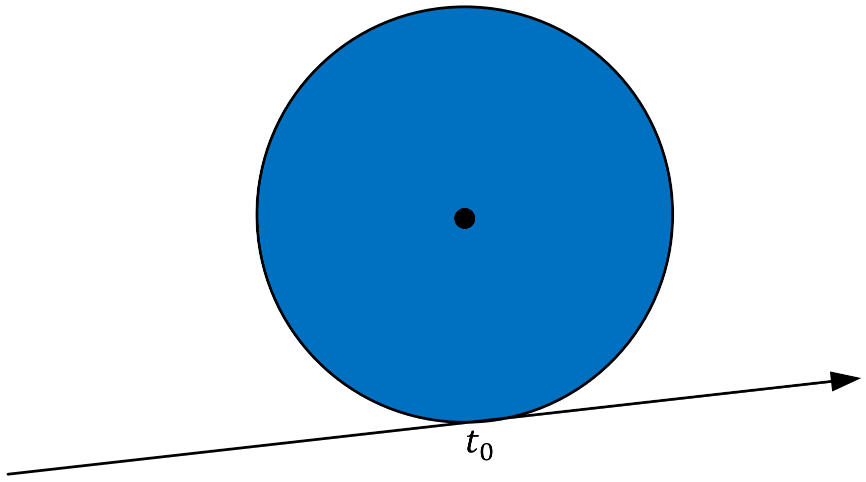


The line hits the sphere in 2 spots

Ray vs. Sphere – Quadratic Equation

Case 3: $\Delta = 0$

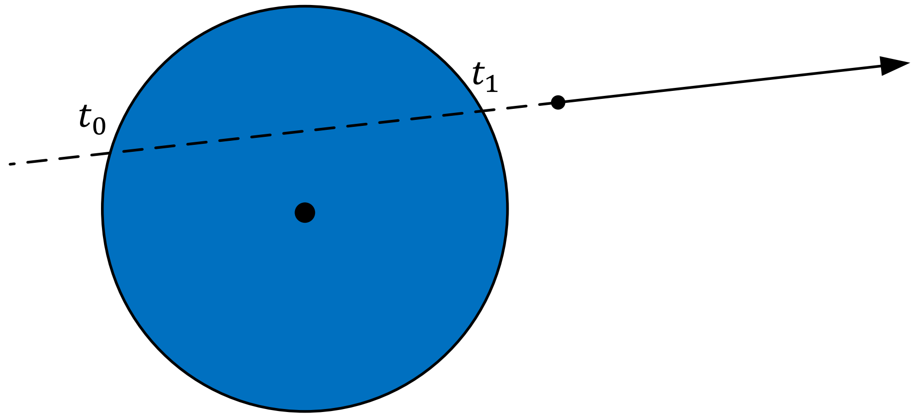
There is only 1 solution



The line is tangent to the sphere

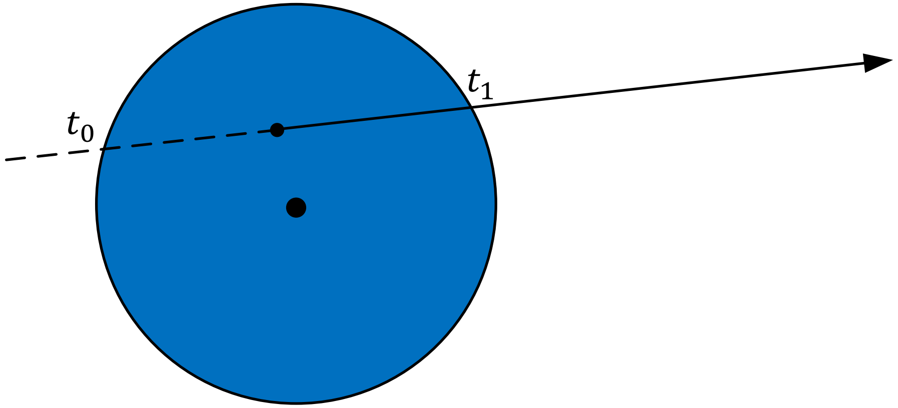
Ray vs. Sphere – Invalid t-values

Important! $\Delta \geq 0$ does not guarantee a "correct" t-value!



Both t-values are invalid:

no intersection

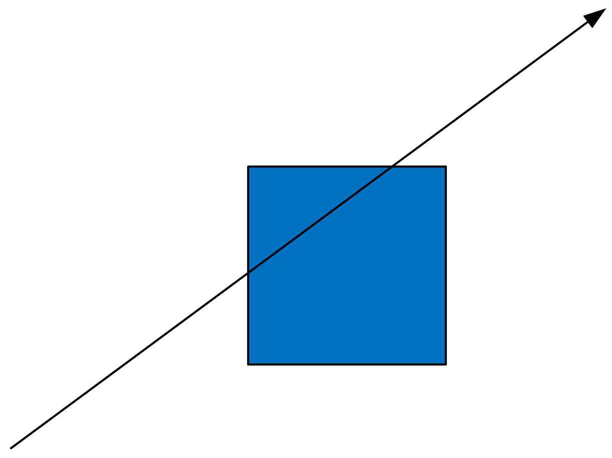


The ray starts inside the sphere. T should be 0.

A t-value can be behind the ray! All negative t-values are invalid!

Ray vs. Aabb

There's no equation for an Aabb



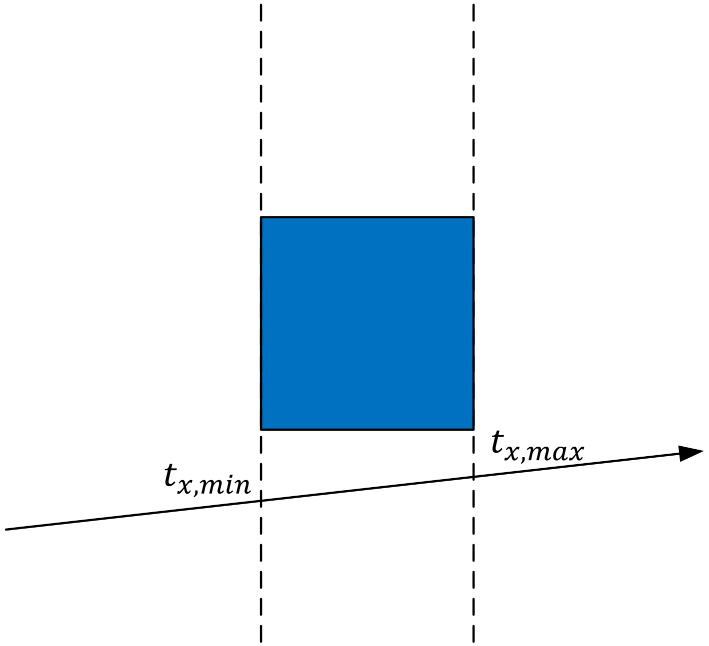
Perform each axis test independently

Combine the results afterwards

Ray vs. Aabb

Each axis has 2 planes

Need to compute a min/max range for each axis



For the x-axis:

$$\vec{n} \cdot \left(\vec{s}_r + \vec{d}_r t_{x,max} - \vec{p}_{max}\right) = 0$$

$$\vec{n} \cdot \left(\vec{s}_r + \vec{d}_r t_{x,min} - \vec{p}_{min}\right) = 0$$

$$\vec{n} = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}$$

Don’t do this!

Ray vs. Aabb

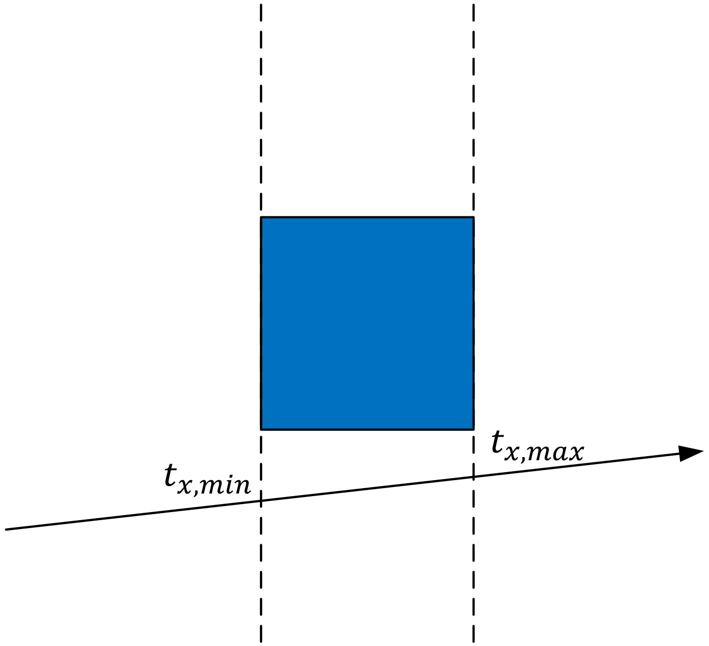
Each axis is independent, why are we using the full vector equation?

Since $\vec{n} = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}$

$$\vec{n} \cdot \left(\vec{s} + \vec{d}t - \vec{p}\right) = 0$$

becomes

$$s_x + d_x t - p_x = 0$$

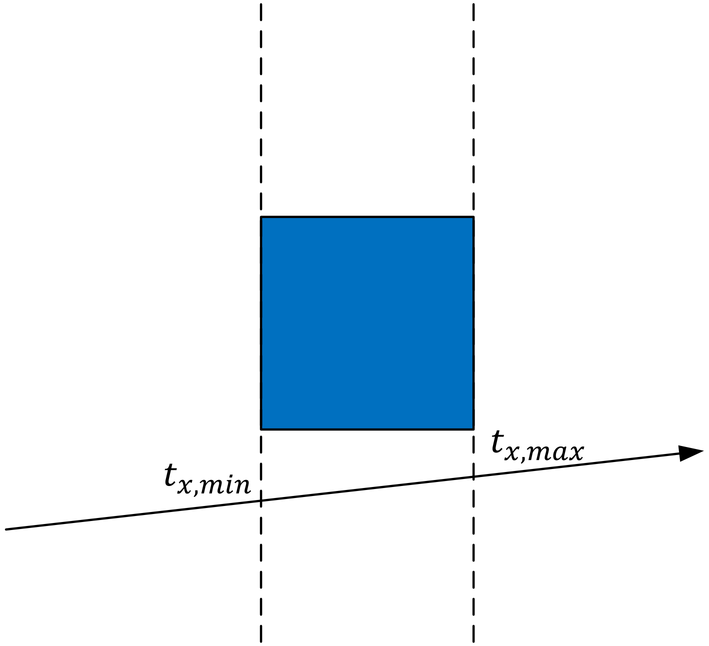


Ray vs. Aabb

Now any axis can define:

$$t_{i,min} = \frac{p_{i,min} - s_i}{d_i}$$

$$t_{i,max} = \frac{p_{i,max} - s_i}{d_i}$$



What problems do we have to consider?

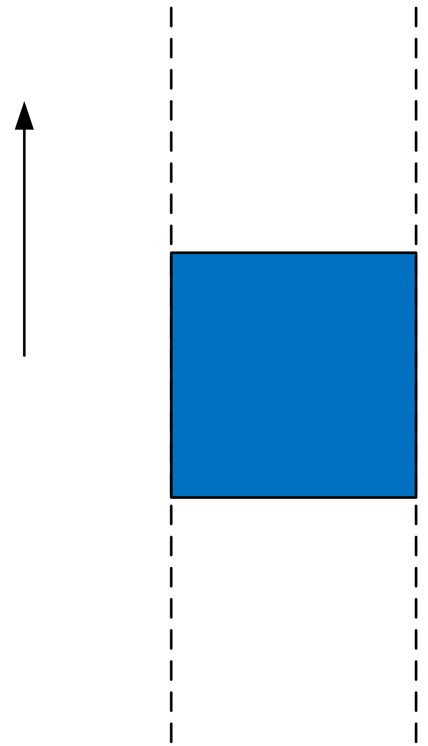
Ray vs. Aabb – Edge Cases

Problem 1: What if $d_i = 0$?

The ray is parallel to the plane

Case 1: The ray might be outside the aabb

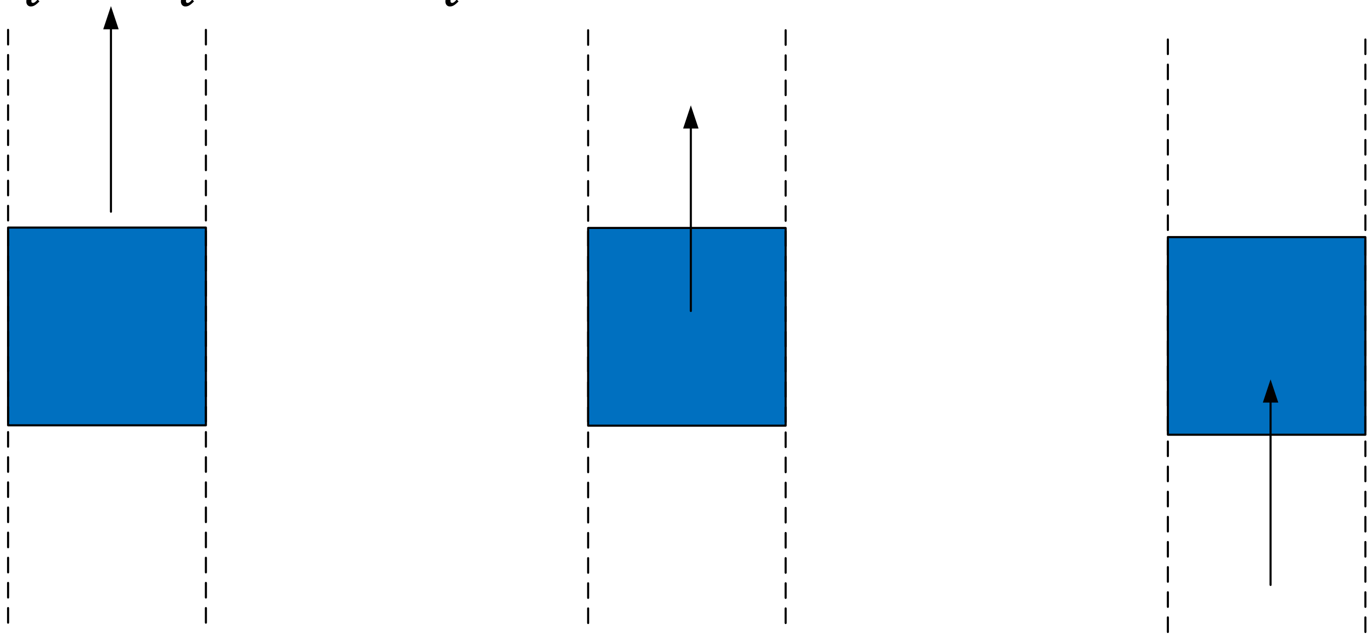
$$s_i < min_i \text{ or } max_i < s_i$$



Ray vs. Aabb – Edge Cases

Case 2: The ray is inside the aabb

$$min_i \leq s_i \leq max_i?$$



We can’t tell from this axis alone

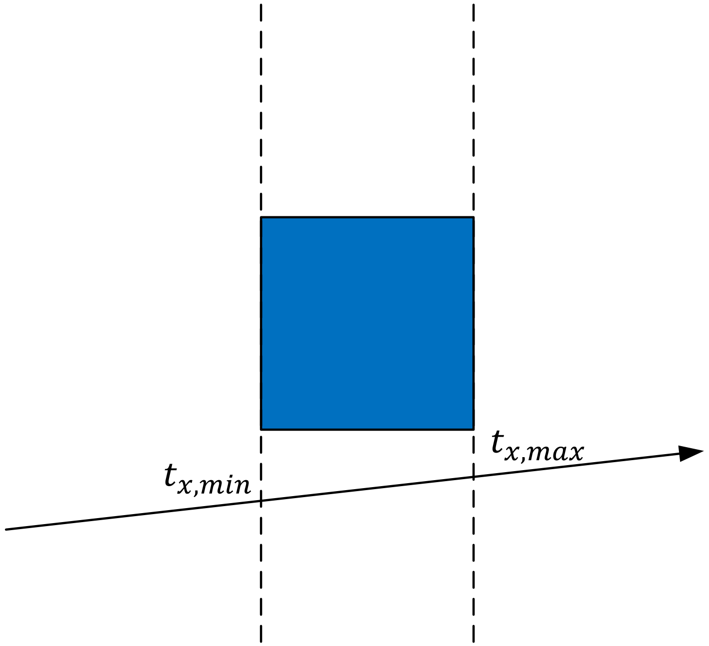
Skip this axis and defer to the remaining axes

Ray vs. Aabb – Edge Cases

Problem 2: Are t_{min} and t_{max} always right?

$$t_{i,min} = \frac{p_{i,min} - s_i}{d_i}$$

$$t_{i,max} = \frac{p_{i,max} - s_i}{d_i}$$



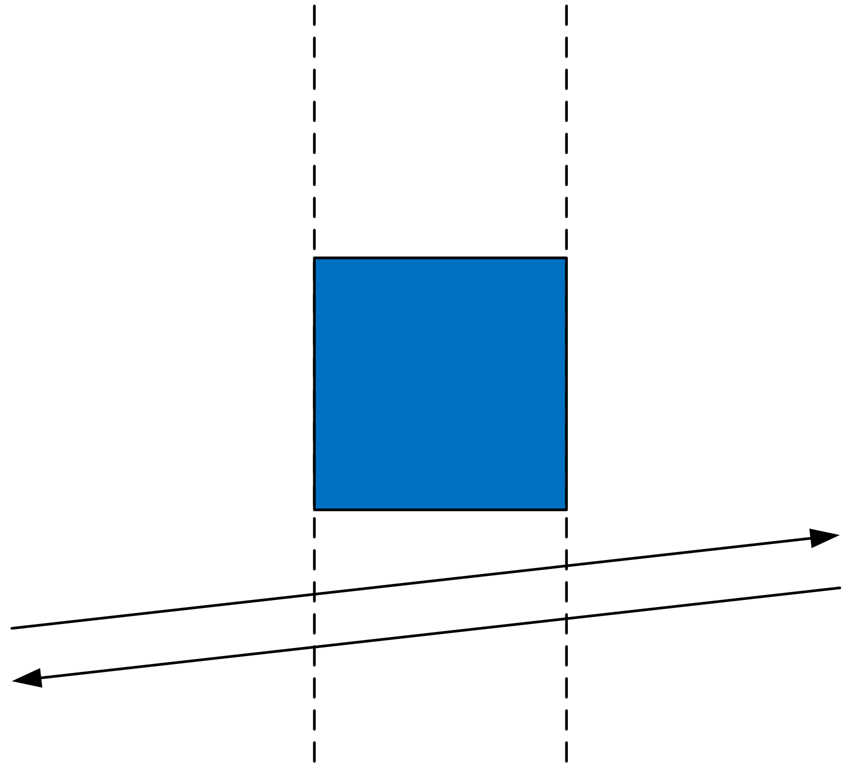
Is there ever a case where this is wrong?

Ray vs. Aabb – Edge Cases

Consider the ray's direction

$$\vec{d}_i > 0 \quad \begin{cases} t_{min} = t(min_i) \\ t_{max} = t(max_i) \end{cases}$$

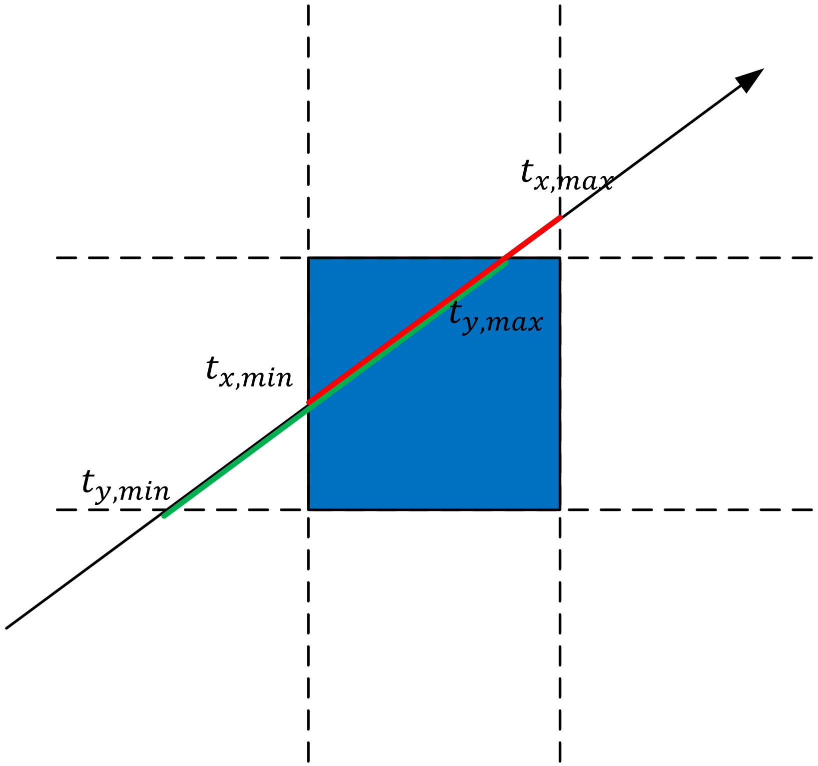
$$\vec{d}_i < 0 \quad \begin{cases} t_{min} = t(max_i) \\ t_{max} = t(min_i) \end{cases}$$



Ray vs. Aabb

Now we have all the axis results

How do we them together?

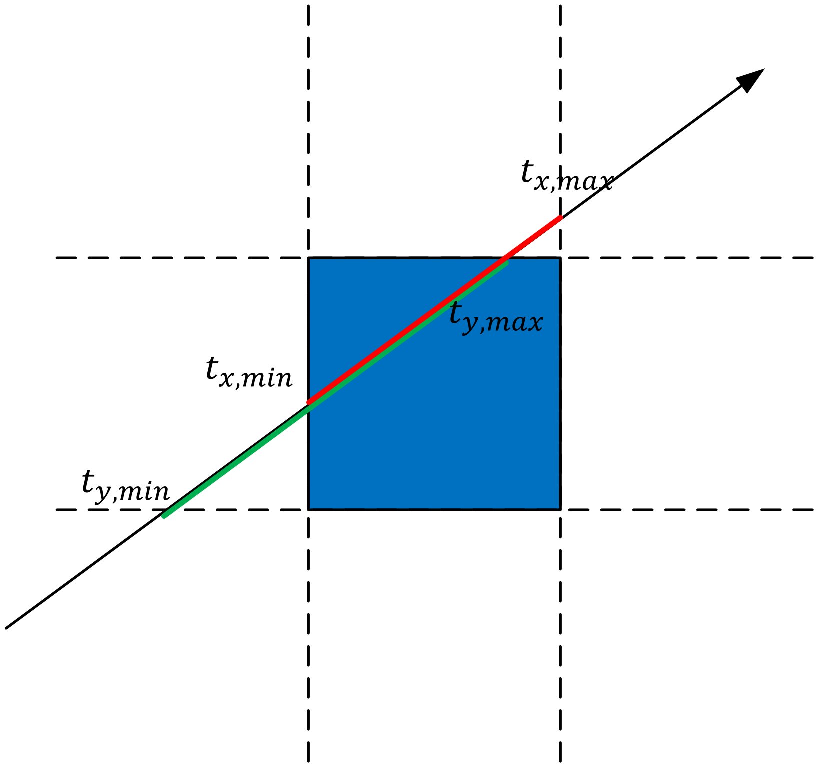


Ray vs. Aabb

We want the last min and the first max t-values

$$t_{min} = \max(t_{i,min}) = t_{x,min}$$

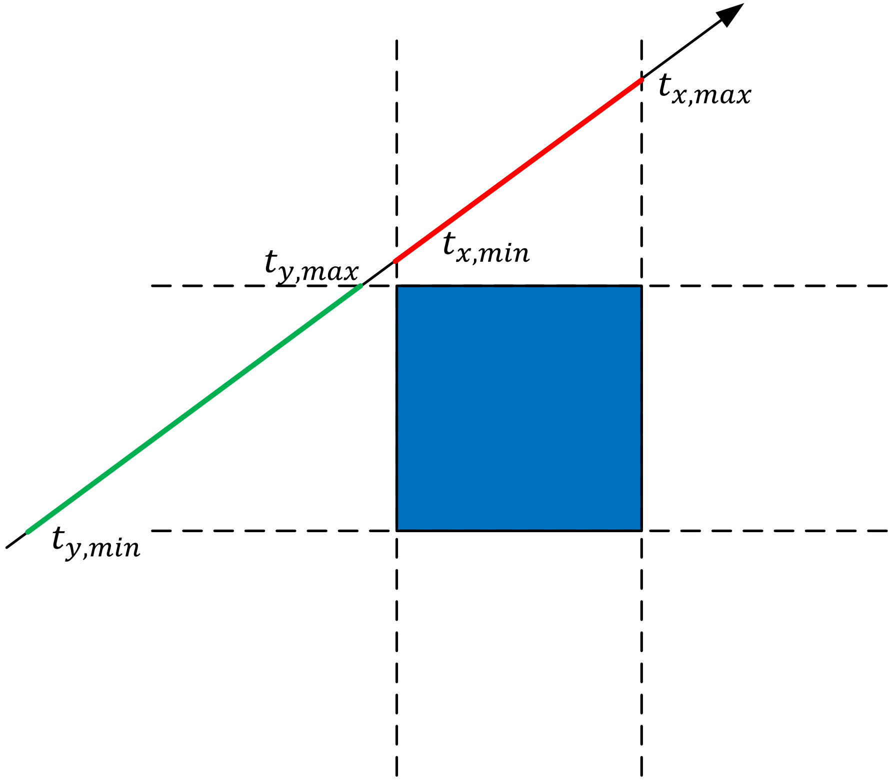
$$t_{max} = \min(t_{i,max}) = t_{y,max}$$



Ray vs. Aabb

What happens when there's no intersection?

$$t_{min} = t_{x,min}$$
$$t_{max} = t_{y,max}$$

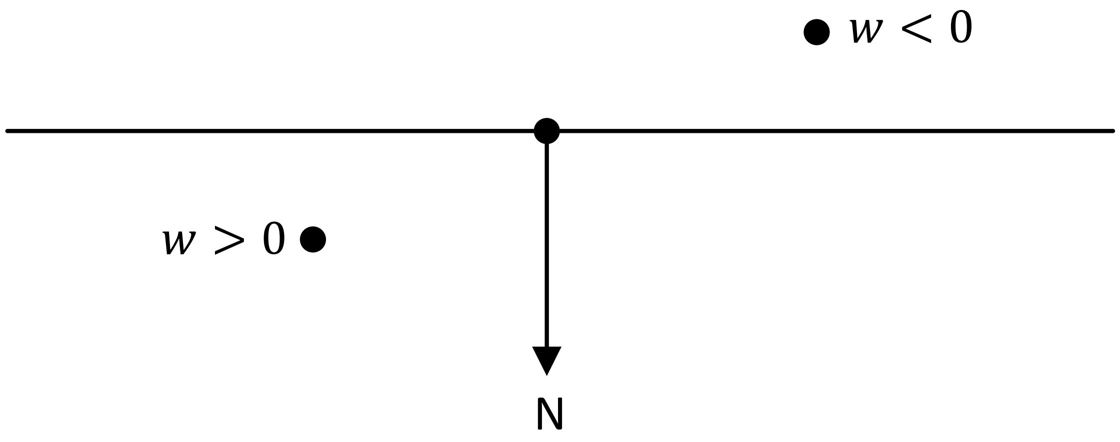


$t_{min} > t_{max} \Rightarrow$ no intersection

Plane vs. Point

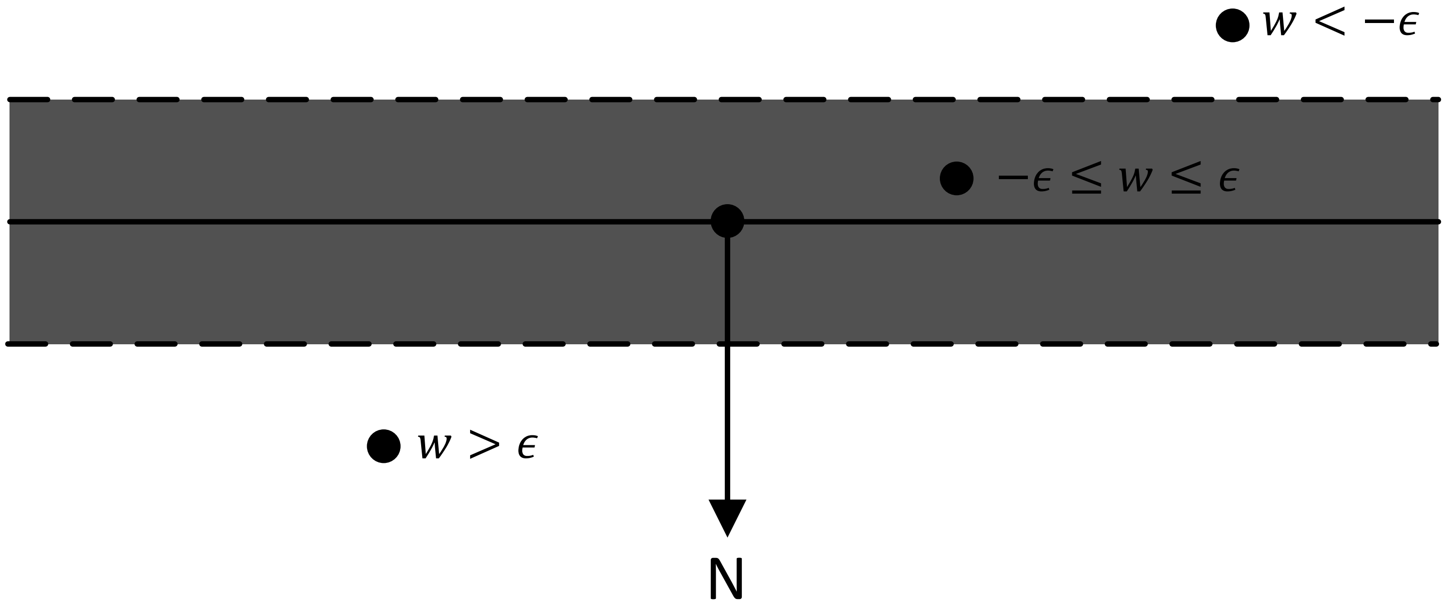
Compute the distance from the plane:

$$\vec{n} \cdot \vec{p} - d = w \qquad \text{or} \qquad \begin{bmatrix} n_x \\ n_y \\ n_z \\ d \end{bmatrix} \cdot \begin{bmatrix} p_x \\ p_y \\ p_z \\ -1 \end{bmatrix} = \mathrm{w}$$



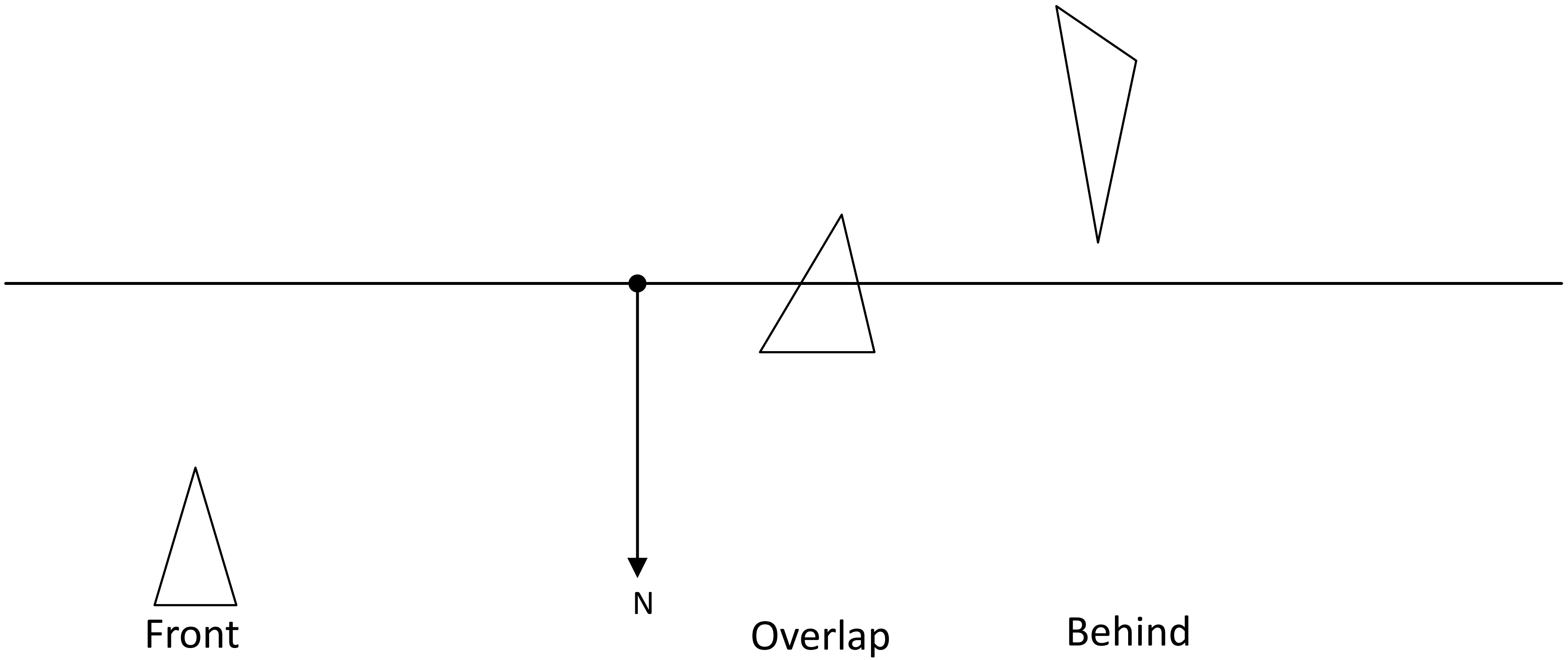
Plane vs. Point

What about numerical robustness (thick planes)?



Plane vs. Triangle

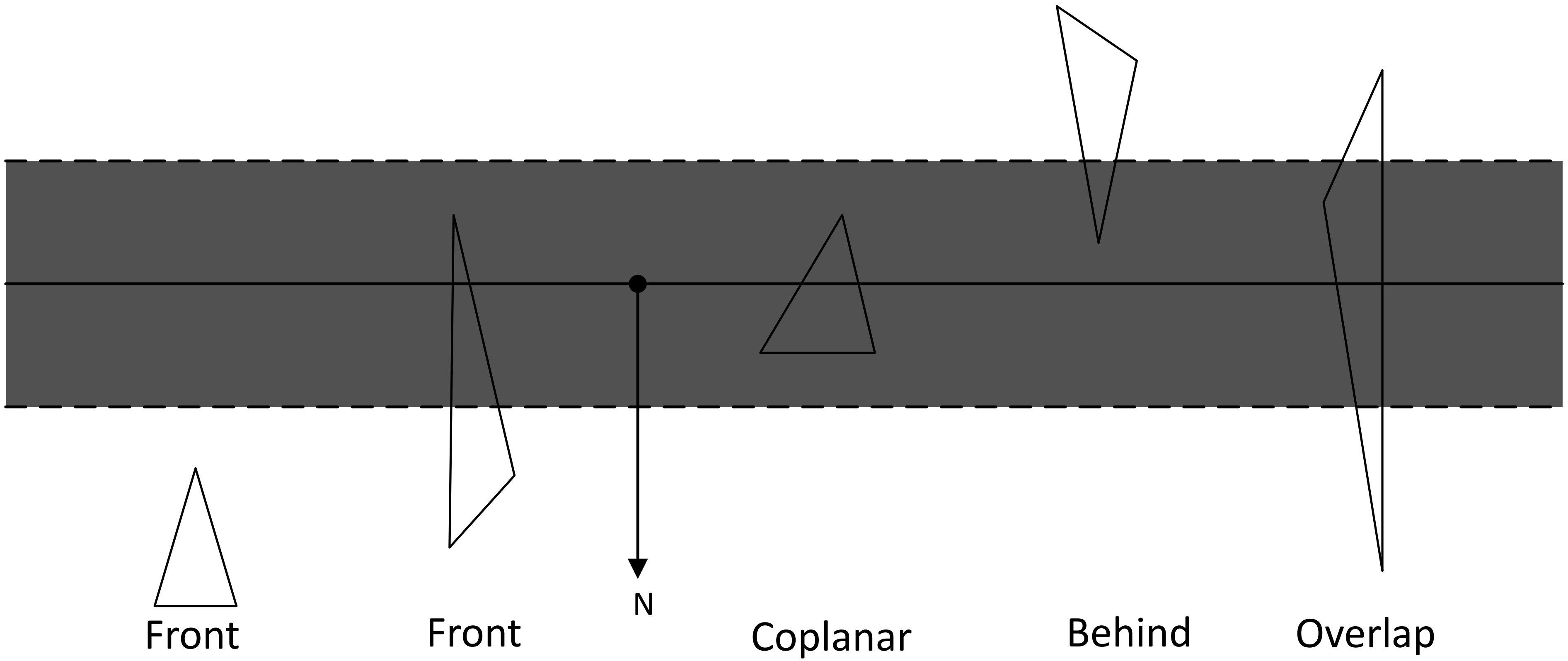
Combine the results of Point vs. Plane for all the points?



What are we missing?

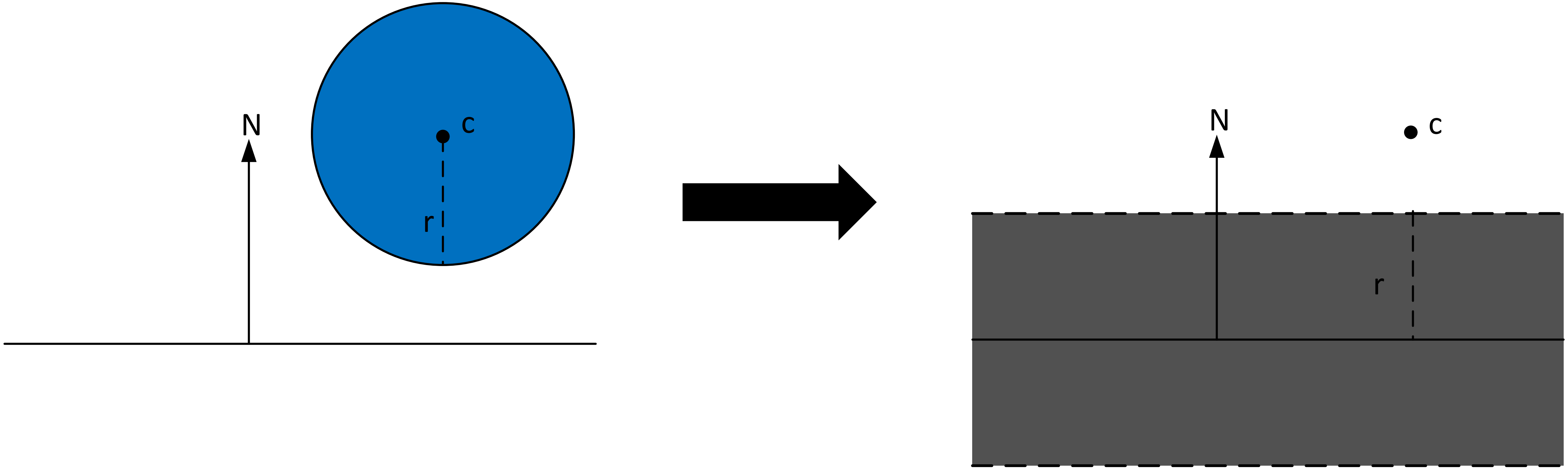
Plane vs. Triangle

We have to consider thick planes (epsilon)



Plane vs. Sphere

Conceptually turn Plane vs. Sphere into Plane vs. Point



Plane vs. Aabb

Method 1: Classify all points against the plane

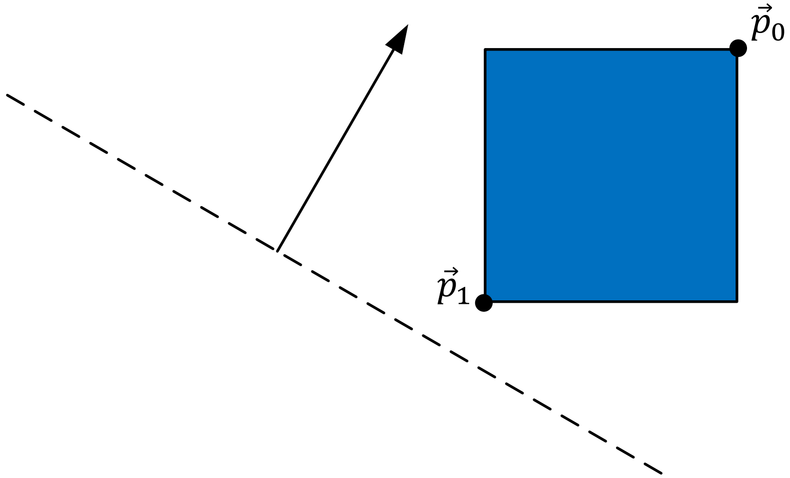
All in-front: Aabb in front

All behind: Aabb behind

Otherwise: Overlaps plane

Plane vs. Aabb

Method 2: Classify the extremal points



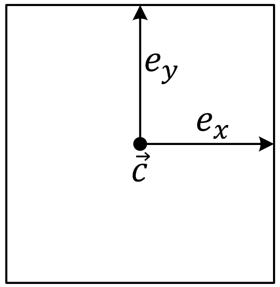
Only two points actually need to be tested

How do we compute these points?

Plane vs. Aabb

How can we find the point furthest in a direction?

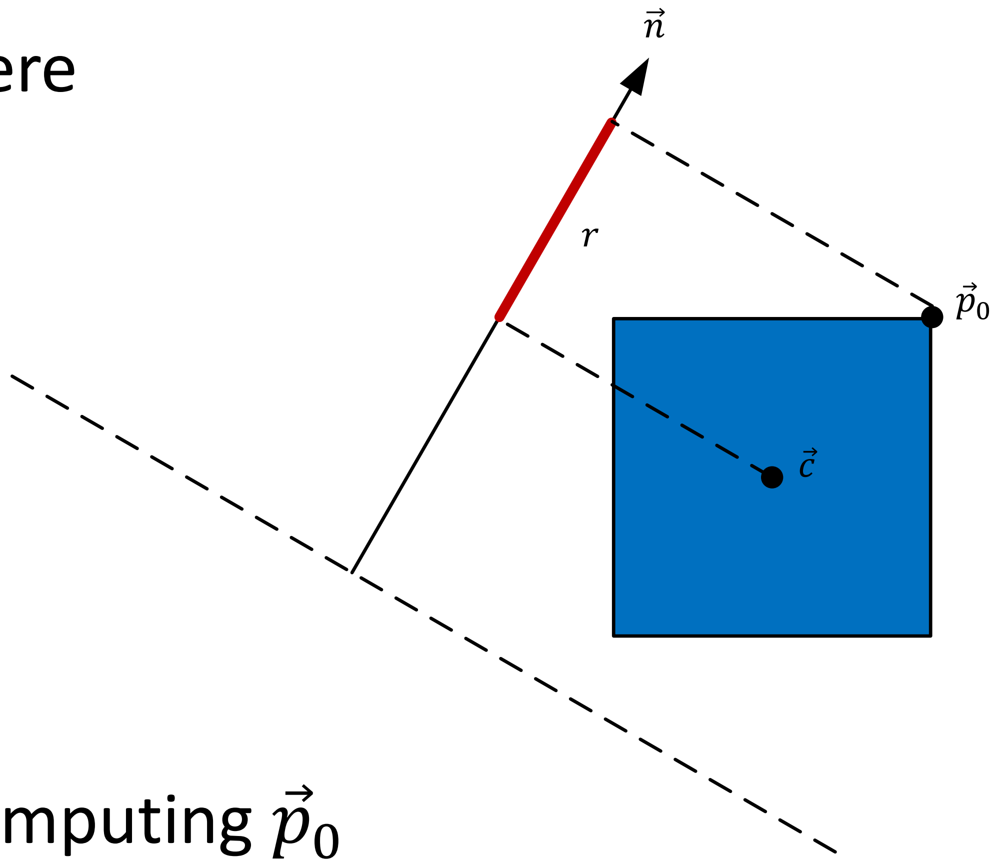
All points can be computed from the center and half-extents



Can determine $+$ or $-$ based upon sign of the vector

Plane vs. Aabb

Method 3: Turn into Plane vs. Sphere



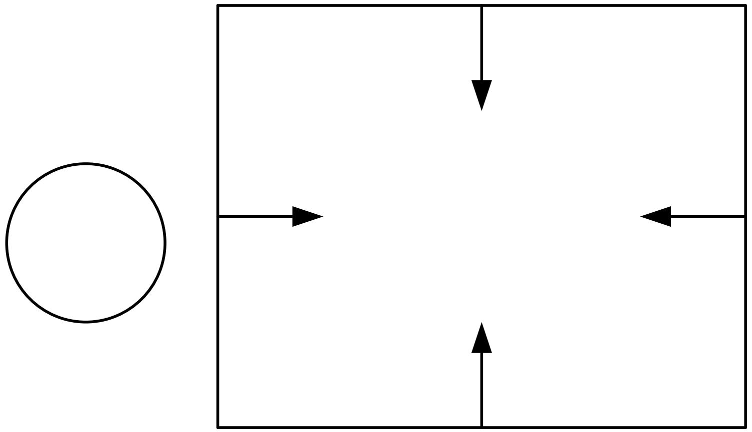
Aabbs are symmetric

A “radius” can be defined

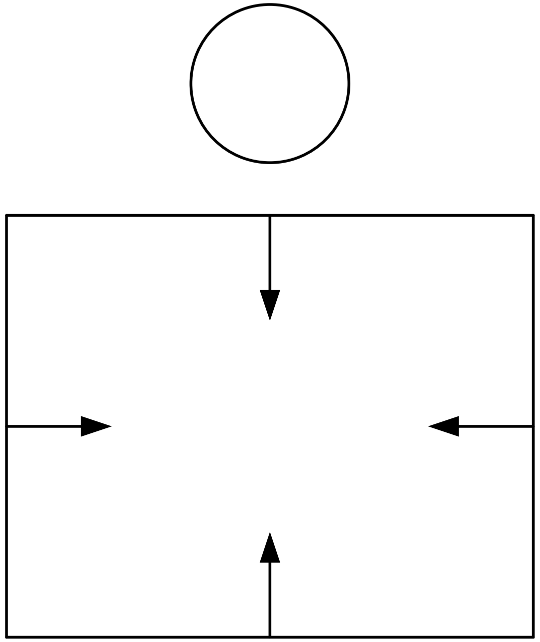
Can compute r directly without computing $\vec{p}_0$

Frustum vs. Sphere Culling

Test all 6 planes:



Outside left plane

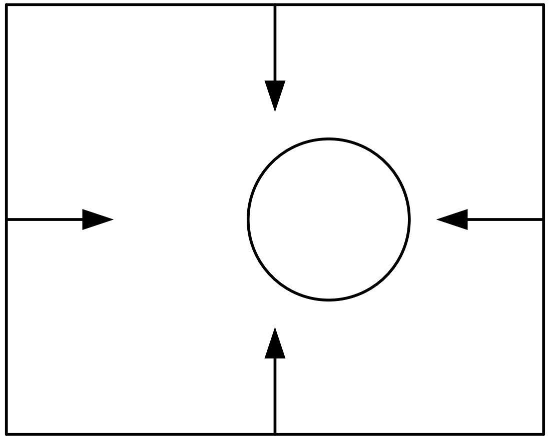


Outside top plane

If the sphere is outside any plane then it is outside the frustum

Frustum vs. Sphere Culling

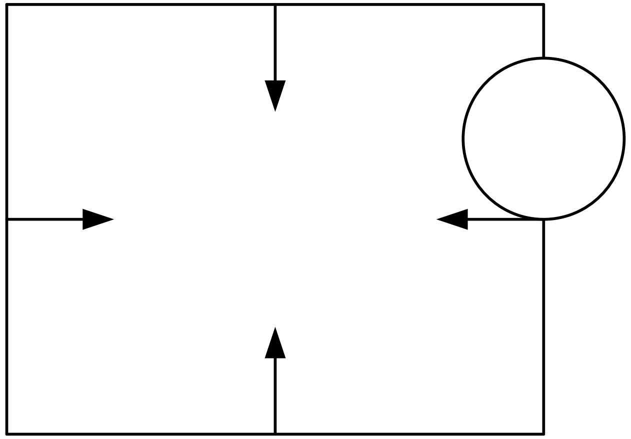
Test all 6 planes:



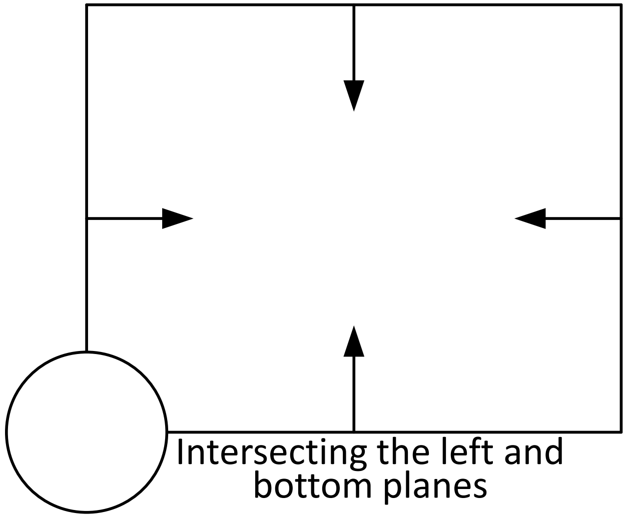
If the sphere is inside all planes then it is inside the frustum

Frustum vs. Sphere Culling

Test all 6 planes:



Intersecting the right plane

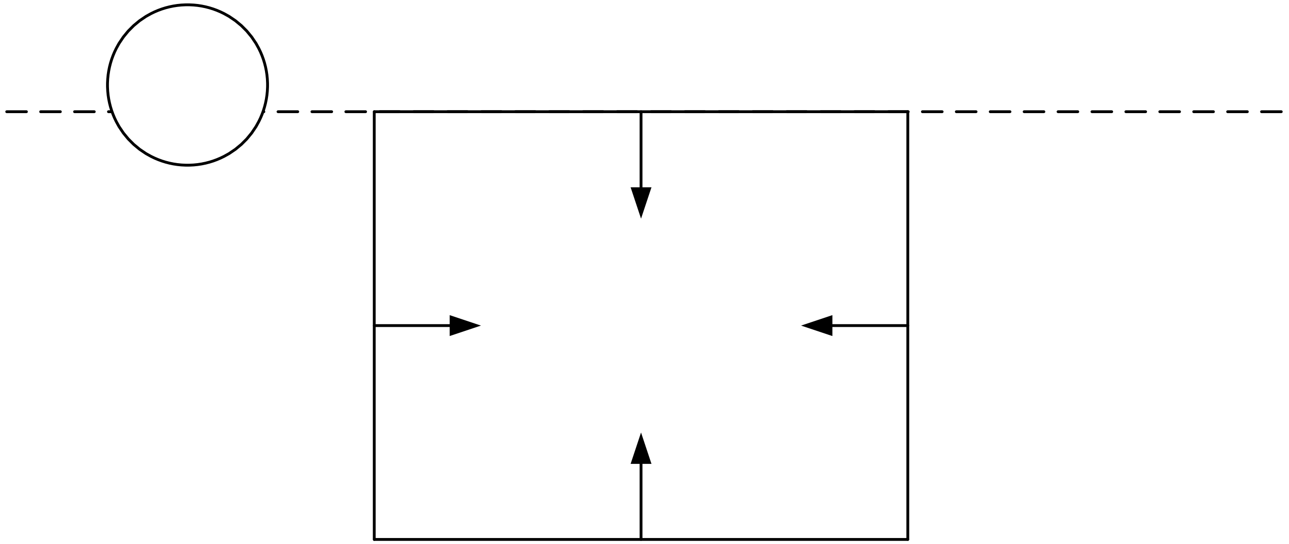


Intersecting the left and bottom planes

Otherwise if the sphere overlaps any plane then it overlaps the frustum

Frustum vs. Sphere Culling

Note: Overlap on one plane does not guarantee an Overlap!!



Intersecting the top plane but outside the left plane

Frustum vs. Aabb Culling

Same as sphere, test all 6 planes:

- If outside any return outside
- If inside all return inside
- Otherwise return overlaps

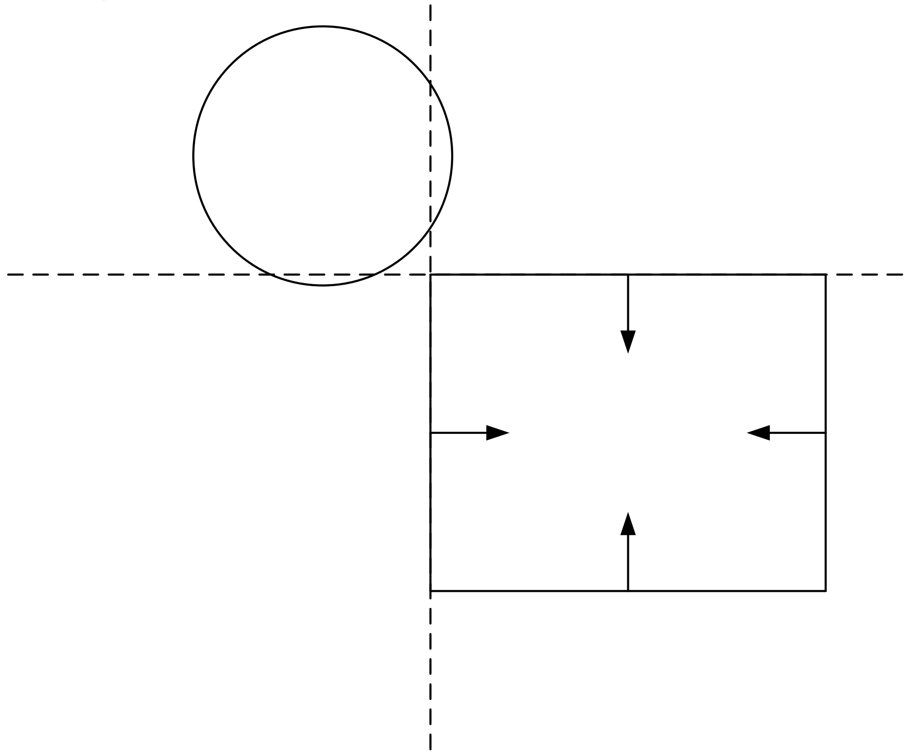
Frustum Culling vs. Frustum Intersection

Frustum Culling is an approximation, it gives false positives

Can you think of a case where Frustum vs. Sphere returns the wrong answer?

Frustum Culling – False Positives

This case returns Overlap when it should return outside



Note: the sphere is not outside any plane!

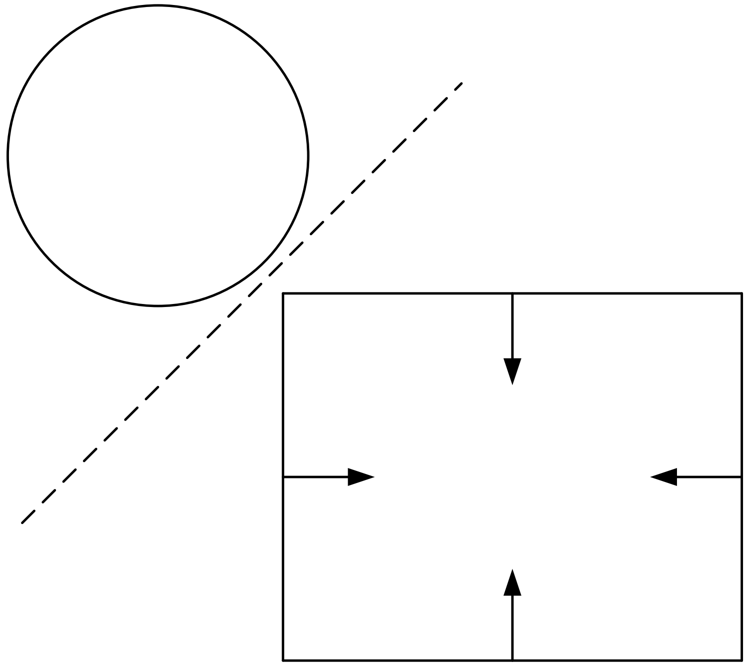
What’s missing? Quick look at SAT

Some extra “planes” need to be tested for correctness

Which planes? Well it depends…

Proper solution is defined by SAT (more later)

Basically, if you can draw a line between them they don’t intersect



Frustum Culling vs. Frustum Intersection

Why not define the proper intersection test?

- More complicated to write
- More computationally expensive
  - Sphere needs 1 more test
  - Aabb needs a total of 26 tests…yes…26

When only culling this is good enough (basic optimizations)

When to use the proper test?

When the exact answer matters! (Picking, etc…)

Frustum Culling – Temporal Coherence

Once we hit a plane that is outside we can return

- Best case only 1 plane test
- Worst case 6 tests

Temporal Coherence: Objects don’t move much from frame to frame

- We can test the planes in any order
- Start with the last plane that returned outside!