Johns Hopkins
Engineering for Professionals

605.767 Applied Computer Graphics

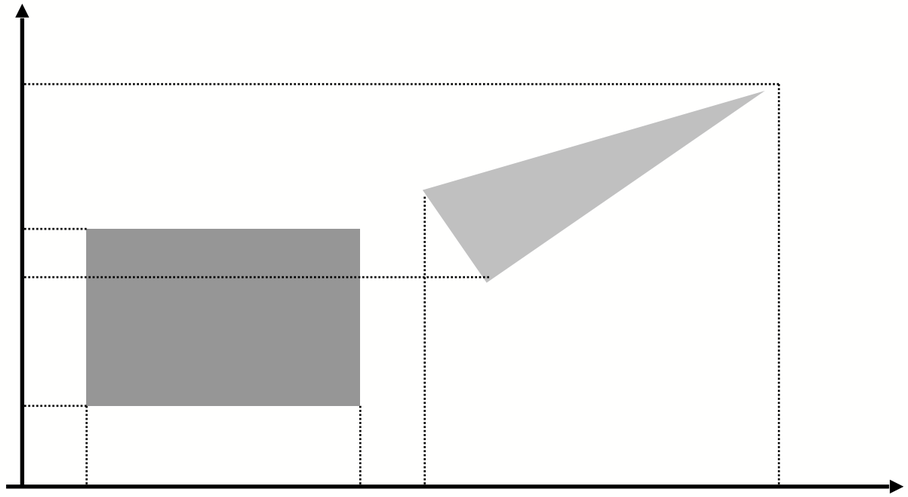
Brian Russin

Module 10C
BV Intersections

Triangle/Box Intersection

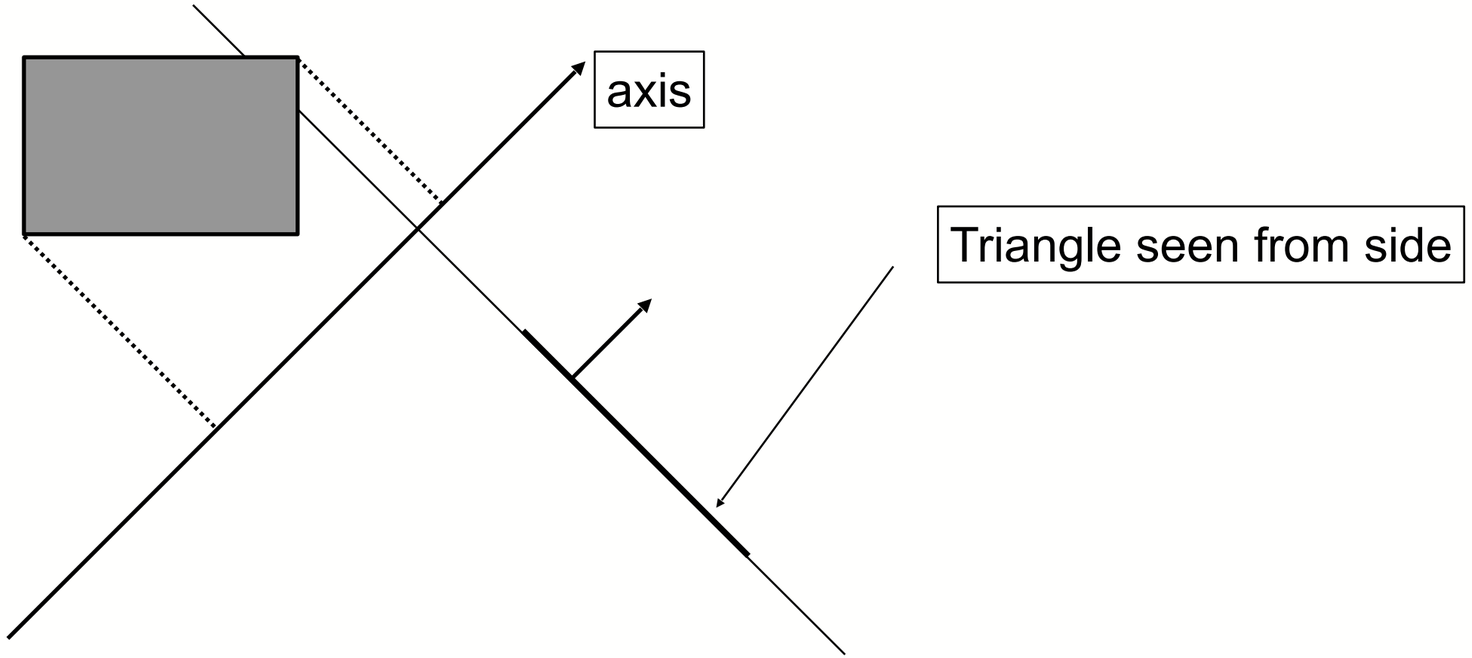
- Detailed description
  - https://fileadmin.cs.lth.se/cs/Personal/Tomas_Akenine-Moller/code/tribox_tam.pdf
- Tomas Akienne-Moller also has some intersection code available:
  - https://fileadmin.cs.lth.se/cs/Personal/Tomas_Akenine-Moller/code
- Test uses the Separating Axis Theorem
  - Method involves 13 axis tests
  - Exit with no intersection as soon as a separating axis is found
- Step 1
  - Perform 3 tests against the axes orthogonal to the faces of the box

A and B overlap on this axis



Triangle/Box Intersection (cont.)

- Step 2
  - Test an axis orthogonal to face of triangle
  - Use fast plane/AABB overlap test
    - Tests 2 vertices of the box diagonal most closely aligned to triangle normal



Triangle/Box Intersection (cont.)

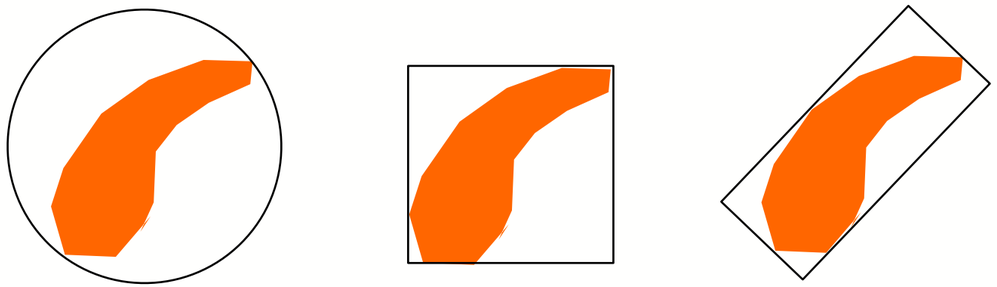
- Step 3
  - Test axis: $a_{ij} = e_i \times f_i$
    - e=normals to box sides (coordinate axes)
    - f=edge of triangle
  - Example
    - x-axis from box: e_{box}=(1,0,0)
    - $f_{triangle}$=v_1-v_0
  - Test all such combinations (9 tests)
  - If there is at least one separating axis, then the objects do not collide
  - Else they do overlap

Triangle/Box Intersection Notes

- Most efficient ordering of the steps (proposed by Moller)
  1. Triangle Edges crossed with box's axes (Step 3)
  2. Box's axes (Step 1)
  3. Triangle plane's normal (Step 2)
- AABB/Triangle test can be optimized
  - Reduction of calculation using principle axes
    - Many values are zero
  - OBB/Triangle cannot be optimized in the same manner

BV/BV Intersection Tests

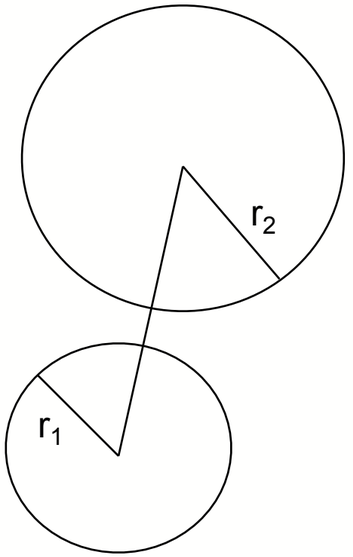
- BV is a closed volume that totally contains a set of objects
- BVs simplify collision detection tests
  - To test if 2 cars collide, first test if the BVs containing the cars collide
  - Sphere and AABB are simple to construct and test
    - More complex BVs like Oriented Bounding Box can provide a better fit
      - Efficiency of BV can be estimated by empty space within BV



OBB provides a better fit!

Sphere/Sphere Intersection

- Simple intersection test
  - Compute distance between sphere centers
  - Reject if distance > sum of the radii
    - Otherwise they intersect
  - Generally use squared distances

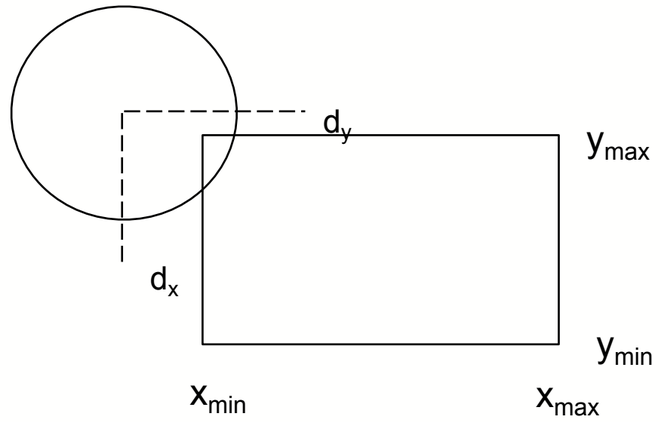



```
/**
 * Test to determine if 2 spheres intersect.
 *
 * @return  Returns true if the 2 spheres overlap/intersect,
 *                false if they are disjoint
 */
bool BoundingSphere::Intersect(BoundingSphere& sphere)
{
  Vector3 l = sphere.center - center;
  return (l.NormSquared() <= SQR(radius + sphere.radius));
}
```

Sphere/AABB Intersection

- Arvo presented a simple method to determine if a sphere intersects an AABB
  - Find the point on AABB that is closest to the sphere center
  - One dimensional tests against each coordinate axis
    - Check sphere center coordinate against AABB
    - If outside the distance along the axis is found and squared
  - Sum of these squared distances is compared to radius squared
- Optimization: exit early if distance along any axis is greater than the radius

```
// Pseudo code
bool BoundingSphere::Intersect(AABB& box)
{
  d = 0;
  for each axis i (x, y, z)
  {
    if (c_i < a_i^min) d += SQR(c_i - a_i^min)
    else if (c_i > a_imax) d += SQR(c_i - a_i^max)
  }
  return (d <= r^2)
}
```



AABB/AABB Intersection

- Intersection of 2 AABBs is trivial
  - Test the min/max of each coordinate axis
    - Disjoint (no intersect) if extents are separate

```
/**
 * Test to determine if 2 AABBs intersect.
 *
 * @return  Returns true if the 2 AABBs overlap/intersect,
 *          false if they are disjoint
 */
bool Intersect(AABB& box)
{
  // Disjoint if min > box.max or max < box.min for any coordinate axis
  return  !(bounds[0] > box.bounds[3] || bounds[3] < box.bounds[0] ||
            bounds[1] > box.bounds[4] || bounds[4] < box.bounds[1] ||
            bounds[2] > box.bounds[5] || bounds[5] < box.bounds[2]);
}
```

OBB/OBB Intersection

- Apply separating axis tests
  - 15 total tests
    - Each of the axes of the two boxes
      - 6 tests
    - Cross product of axis of one box with axis of the other
      - 9 tests
      - Similar to Step 3 of Triangle/Box test
  - If no separating axis is found, the boxes are intersecting
- Expensive compared to simpler bounding volume tests