Separating Axis Theorem for Oriented Bounding Boxes

By: Johnny Huynh
Written on: 12/16/2008
Revised on: 9/15/2009

Note: This tutorial is based on understanding how the Separating Axis Theorem is applied in Gottstalk's "OBB-Tree: A Hierarchical Structure for Rapid Interference Detection."

www.jkh.me

Table of Content

Separating Axis Theorem and Rectangles in 2D Space 3

Computing the Intersection of Two Oriented Bounding Rectangles 12

Separating Axis Theorem and Boxes in 3D Space 17

Computing the Intersection between Two Rectangles in 3D Space is Problematic 27

Computing the Intersection of Two Oriented Bounding Boxes 28

Optimizing the Computation of T•L 31

Optimized Computation of OBBs Intersections 34

Appendix 38

- Case Studies: Detecting Intersecting Boxes 38
- Proof sA • (B × C) = sC • (A × B) 39
- Proof T • (Ax × Bx) = (T • Az)(Ay • Bx) – (T • Ay)(Az • Bx) 41

Bibliography 45

Separating Axis Theorem and Rectangles in 2D Space

Before we begin our tutorial on collision detection using oriented bounding boxes (OBB), it is necessary for us to understand the Separating Axis Theorem.

For simplicity, we will begin with the 2-dimensional (2D) space before moving on to the 3-dimensional (3D) space.

In 2D space, the Separating Axis Theorem states two *convex* polygons do not intersect if and only if there exists a line such that the projections of the two polygons onto the line do not intersect. The line is known as a separating axis. Consider rectangle A and rectangle B in the illustration below.

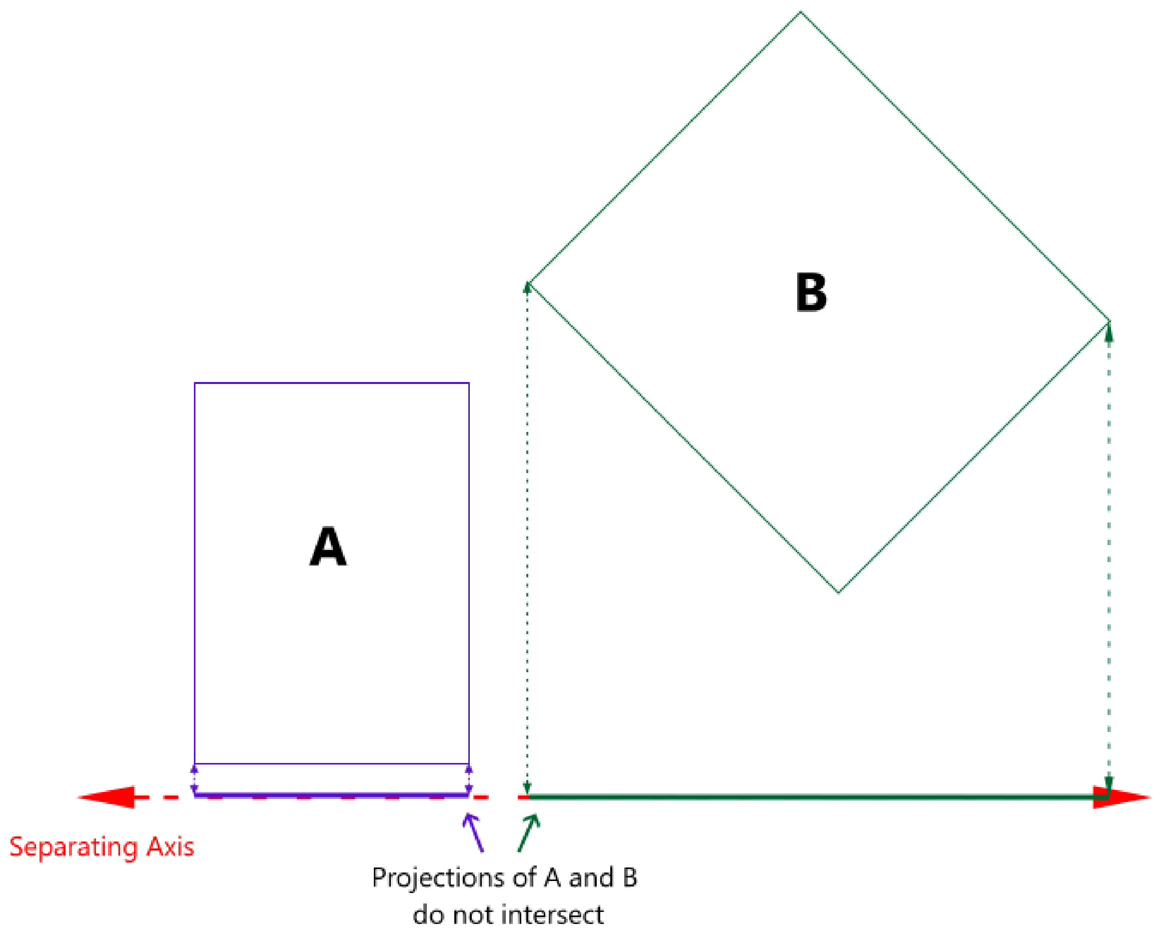


Illustration: Polygons A and B do not intersect because there exists a separating axis such that the projections of A and B onto the axis do not intersect.

The Separating Axis Theorem may also be equivalently defined as two polygons do not intersect if and only if there exists a line that completely divides a polygon on one side of the line and the other polygon on the other side of the line. This line is known as the separating line, which is perpendicular to the separating axis, and is depicted in the illustration below.

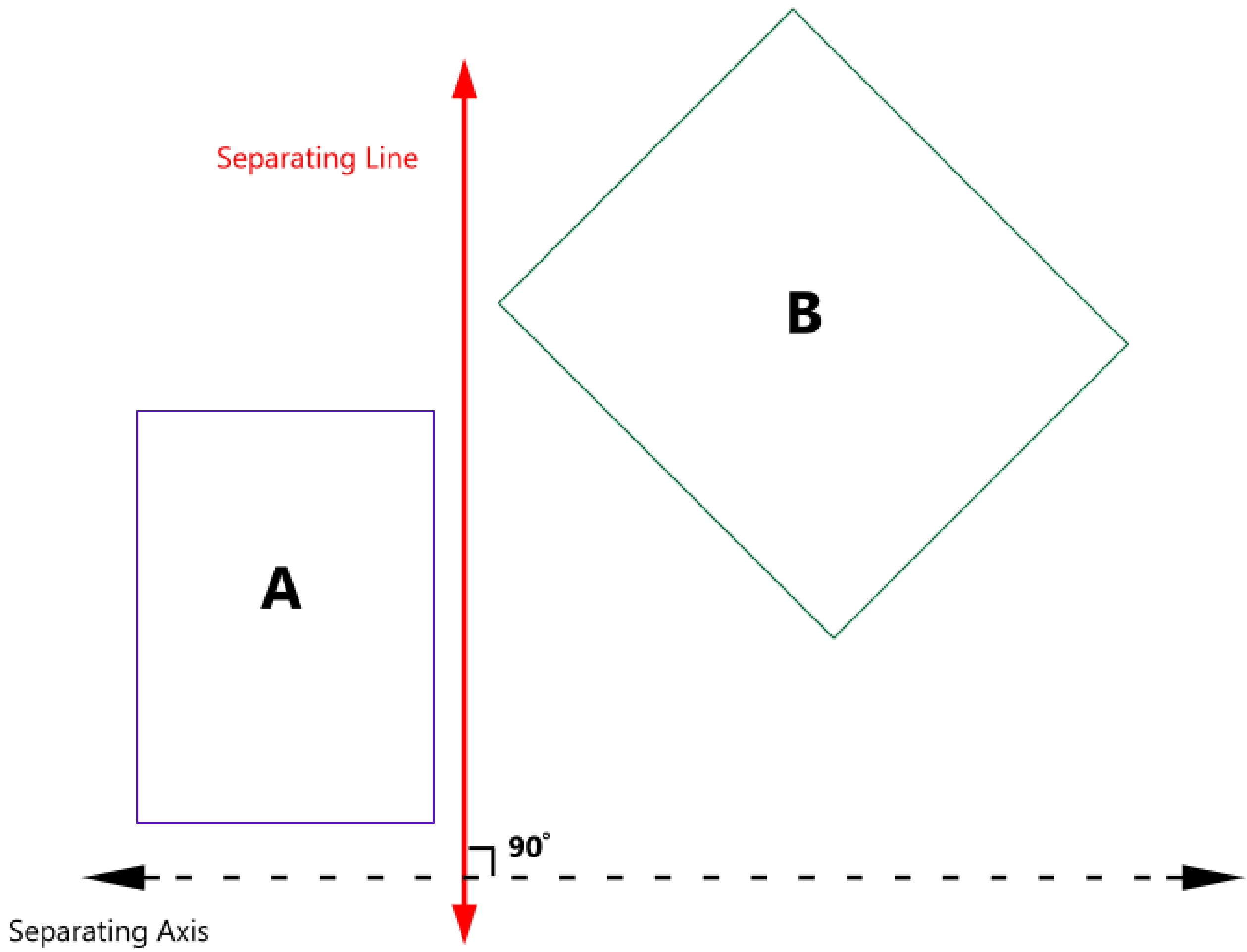


Illustration: Rectangle A and rectangle B are divided by a line, also known as a separating line. The separating line is perpendicular to the separating axis.

Note that there may be more than one separating line parallel to one another, but we will generally only consider each set of parallel separating lines as one separating line for simplicity. See the illustration below.

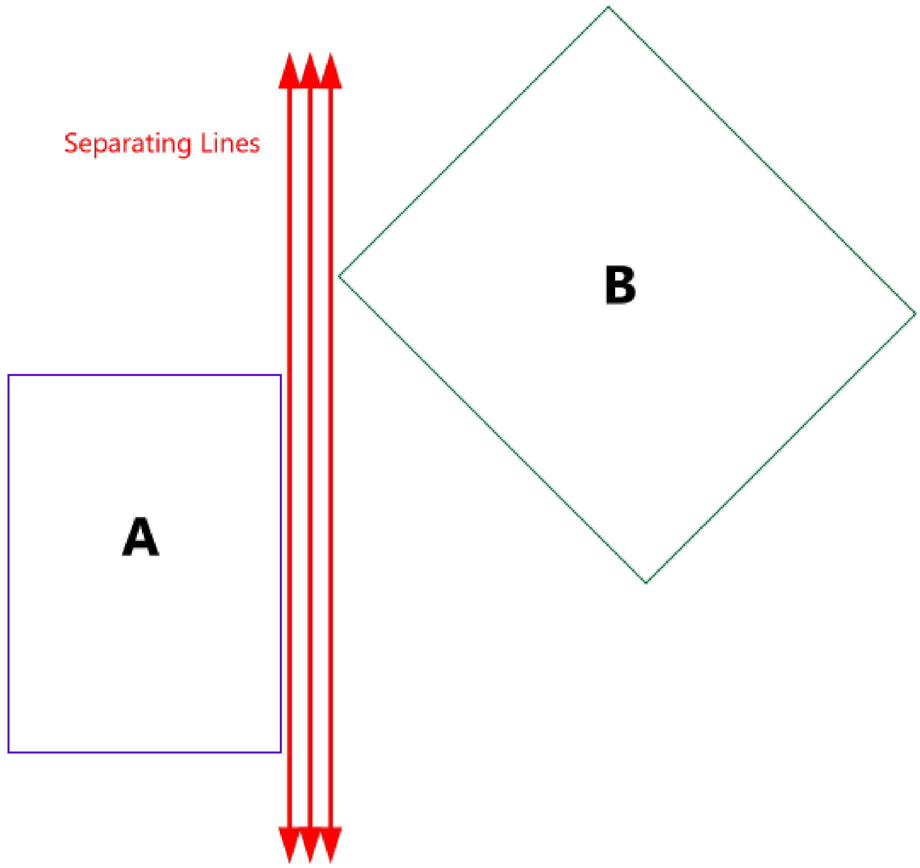


Illustration: We will consider each set of parallel separating lines as one separating line for simplicity.

We need only consider separating lines perpendicular to separating axes because a separating line exists if and only if there exists a separating line that is parallel to an edge of rectangle A or B. In other words, there would have to be at least one separating line that is parallel to an edge of rectangle A or B, if any other separating lines were to exist. In the illustration above, the separating lines are parallel to the height dimension of rectangle A. Hence, to check for an intersection, we will only need to check the lines parallel to the edges of both A and B.

In order to determine whether two polygons intersect, we need to find whether any separating line exists. If a separating line exists, then the two polygons do not intersect; otherwise, the two polygons are intersecting.

Notice a rectangle only has 2 unique axes for its edges because each edge of a rectangle is conveniently parallel to the edge on the other side. Since we know if any separating line is to exist, then the separating line would be parallel to one of the edges (of the two polygons), in checking for a separating line, we only need to consider 2 lines for each rectangle. Therefore, for 2 rectangles, we only need to consider 4 lines. This is illustrated below.

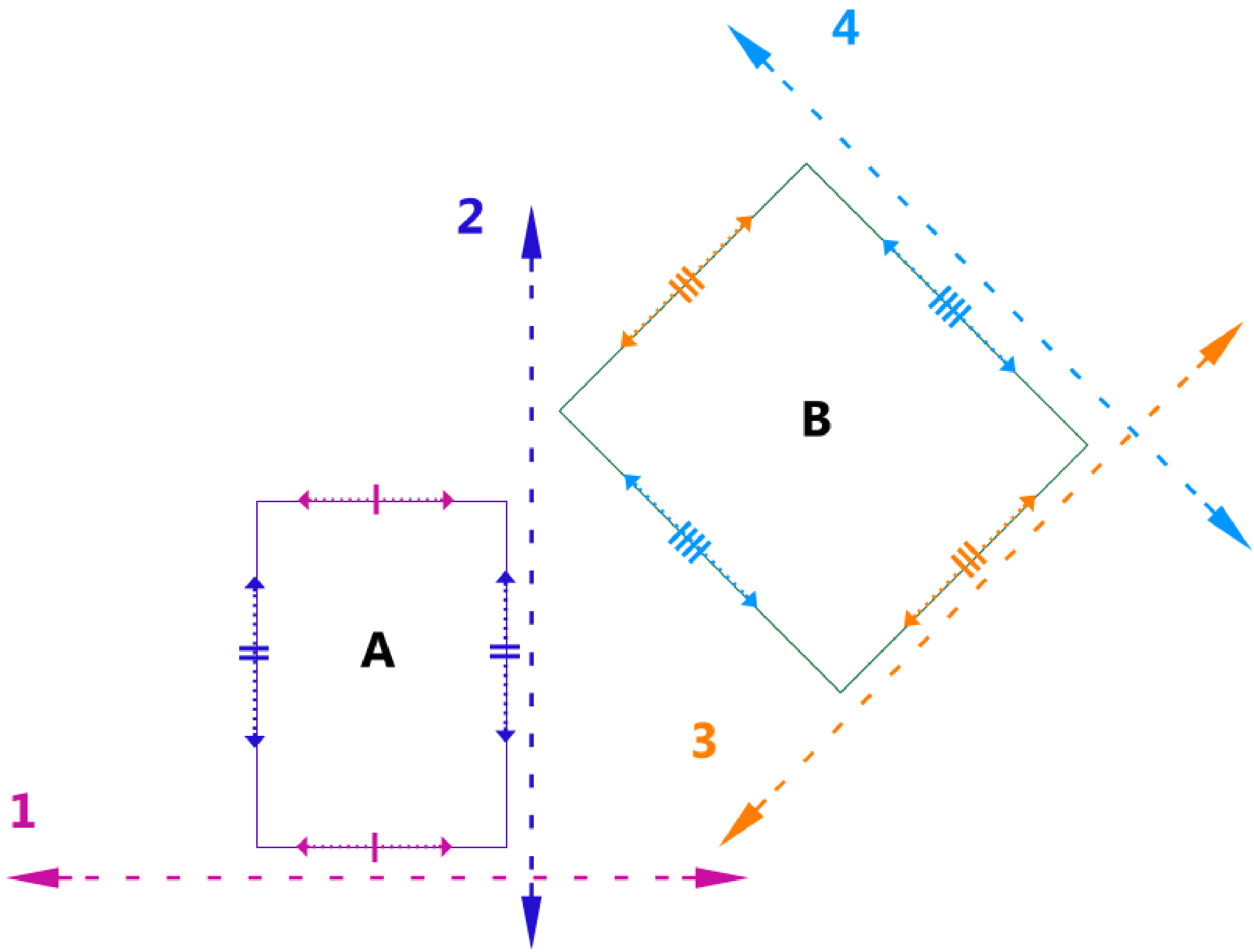


Illustration: To find whether rectangle A and B intersect, we need only consider 4 possible separating lines, which are parallel to the edges of rectangle A and B.

By the Separating Axis Theorem, a separating line exists if and only if there exists a separating axis (i.e. an axis perpendicular to the separating line, such that there exists a line with one end touching polygon A and the other end touching polygon B when polygon A and B are projected onto the axis). See the illustration below.

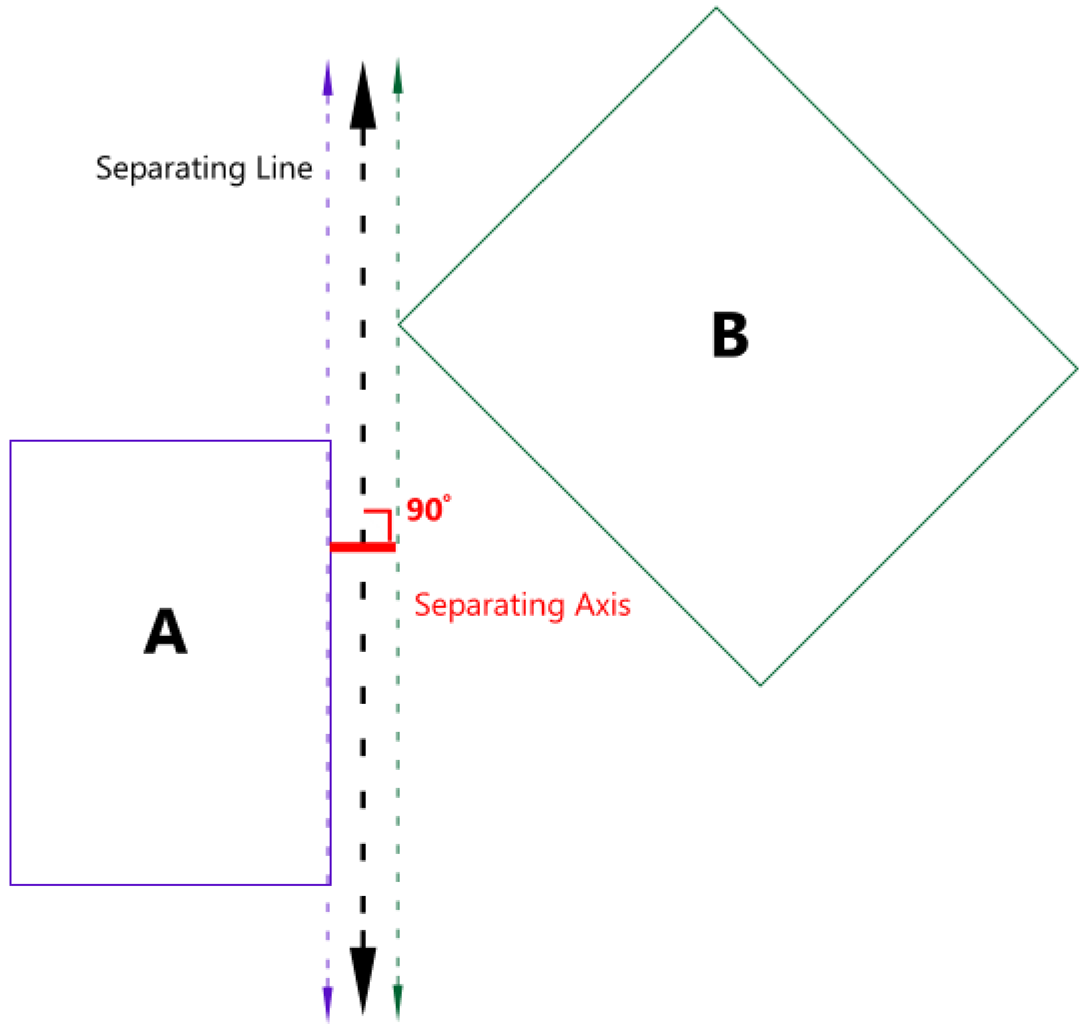


Illustration: A separating line exists if and only if there exists a separating axis such that lines may extend between polygons A and B.

Because a separating line exists if and only if a separating axis exists, we can determine whether a separating line exists by finding whether a separating axis exists.

Conveniently, the axes of the edges of the rectangle are perpendicular to one another. So we can simply project rectangle A and B onto these axes. See the illustration below.

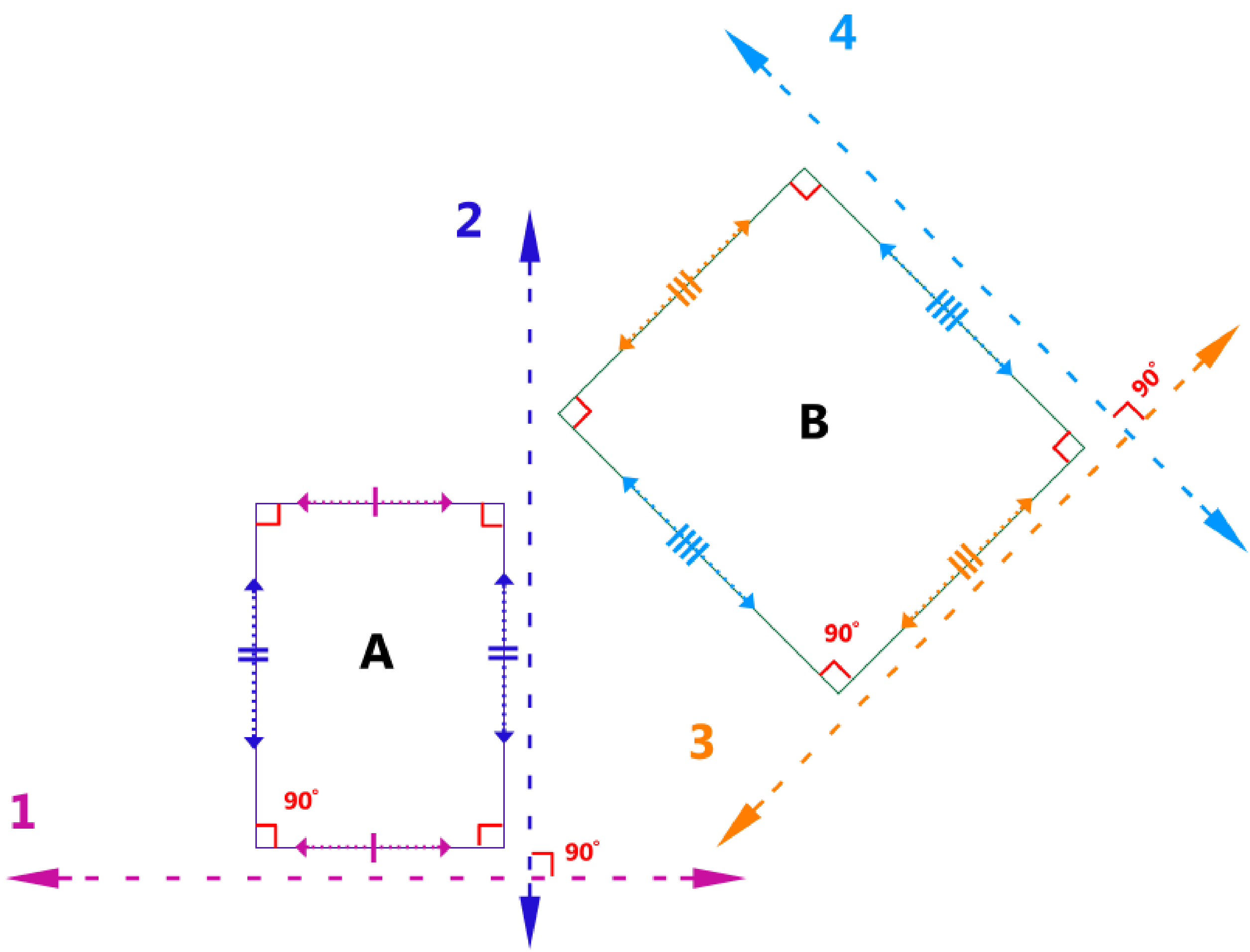


Illustration: Since separating lines exist only when there exists a separating axis that is coincidentally parallel to one of the edges, we need only consider the axes of the edges. This is because the edges of each rectangle are perpendicular to one another. Thus, we can use the axes of the edges to check whether there exists a separating axis.

We will now go through an example. We will use our two rectangles, A and B, as an example. Thus, there are 4 separating lines to check for, and we will find whether any separating line exists by checking whether any of the 4 corresponding separating axes exists. Hence, there are 4 cases to consider.

For case 1, suppose we want to check for a separating line parallel to axis 1, such that the separating line divides rectangle A on one side of the line and rectangle B on the other side of the line. Then we need to check for a separating axis that is perpendicular to axis 1. This is equivalent to checking for a separating axis that is perpendicular to the separating line parallel to axis 1. Conveniently, axis 2 is perpendicular to axis 1, so we can project A and B onto axis 2 to determine whether 2 is a separating axis.

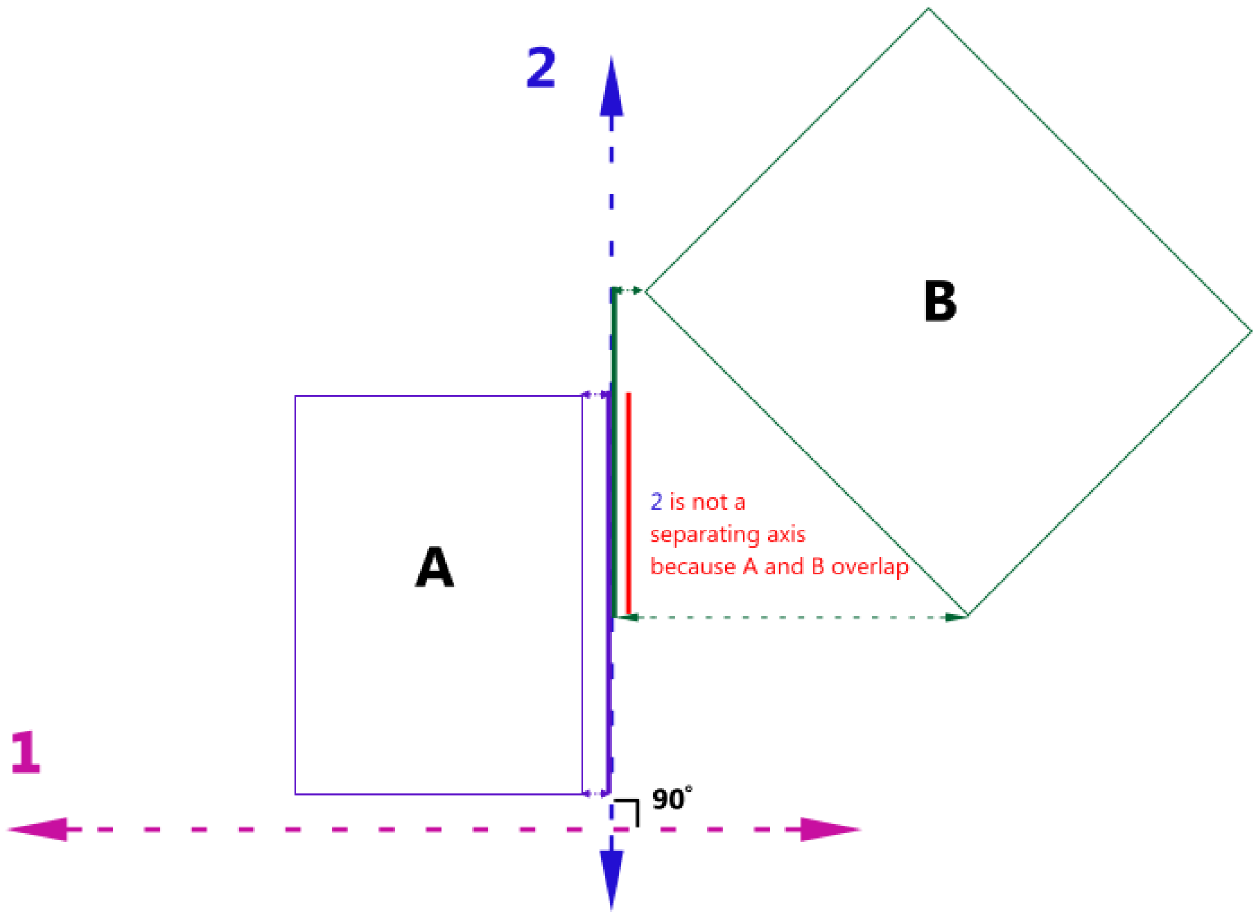


Illustration: The axis 2 is not a separating axis because A and B overlap, which implies there is no separating line parallel to axis 1.

Since we did not find a separating axis, we go to case 2, and suppose we want to find if there exists a separating line parallel to axis 2. Since axis 1 is perpendicular to axis 2, we project rectangles A and B onto axis 1.

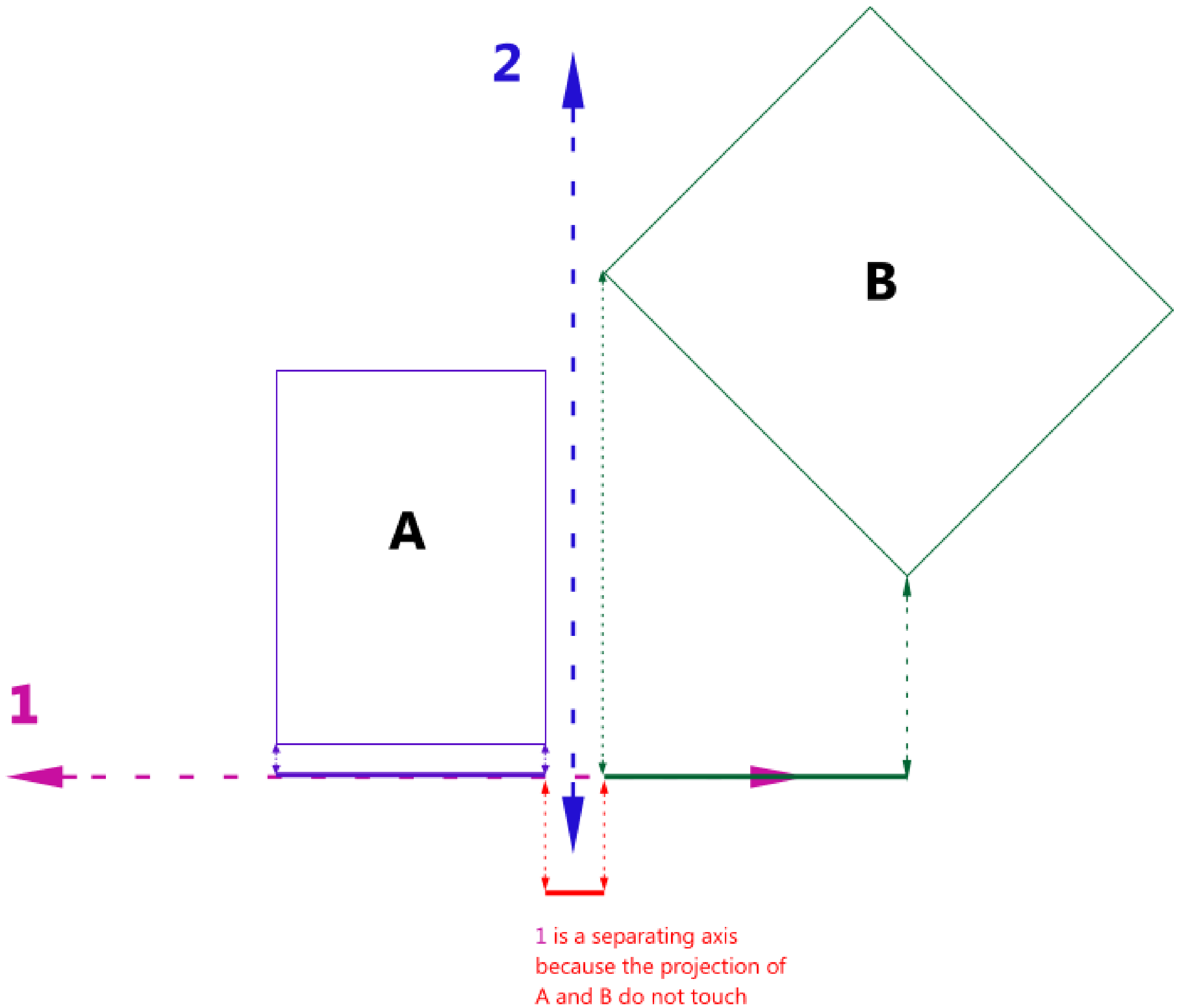


Illustration: The axis 1 is a separating axis because the projections of rectangles A and B onto axis 1 do not overlap nor touch one another, which implies there is a separating line parallel to axis 2.

Since rectangles A and B do not overlap nor touch one another, there exists a separating axis, which means there exists a separating line parallel to axis 2. Hence, we know that rectangles A and B do not intersect, and we are done.

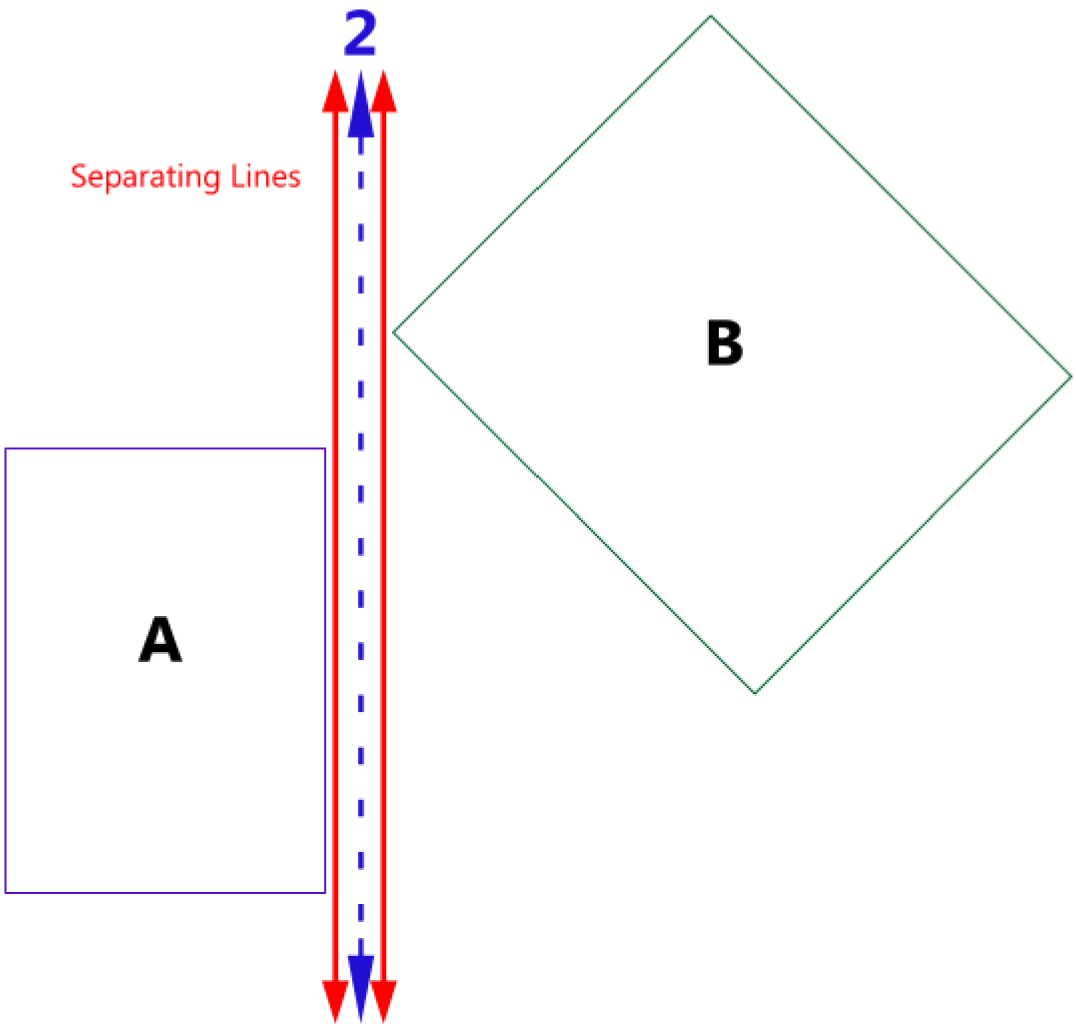


Illustration: There exists separating lines parallel to axis 2.

There is no reason to check for separating lines parallel to axis 3 and axis 4 because we already know that rectangles A and B do not intersect (because there exists at least one separating line). However, if we were to continue, we would find that there is no separating line parallel to axis 3, but there is a separating line parallel to axis 4.

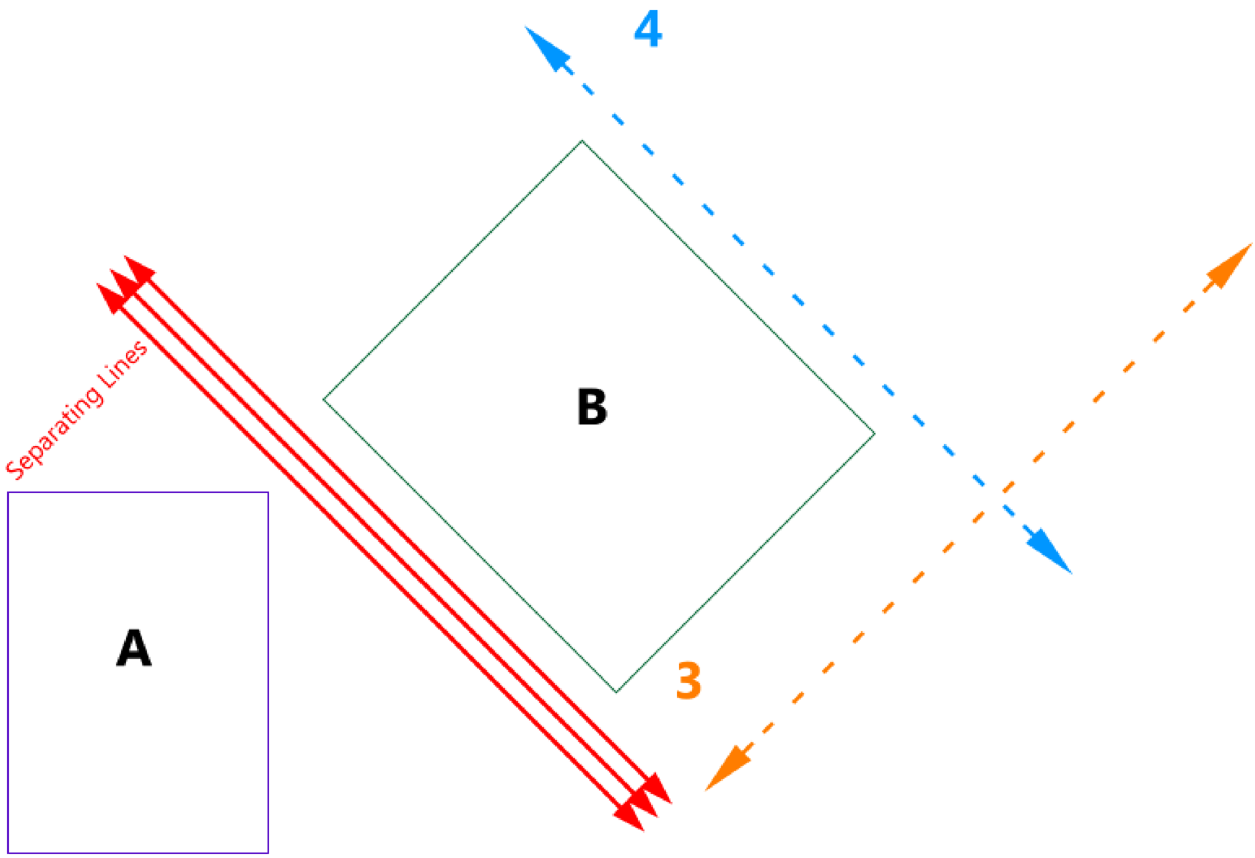


Illustration: There exists separating lines parallel to axis 4, but there is no separating line parallel to axis 3 that separates rectangles A and B.

Computing the Intersection of Two Oriented Bounding Rectangles

Now that we have a better understanding of the Separating Axis Theorem, we will introduce the inequalities used in the computation of finding intersections between any two rectangles.

Suppose we have rectangle A and rectangle B, and we have these properties for the rectangles:

P_A = coordinate position of the center of rectangle A
Ax = unit vector representing the local x-axis of A
Ay = unit vector representing the local y-axis of A
W_A = half width of A (corresponds with the local x-axis of A)
H_A = half height of A (corresponds with the local y-axis of A)

P_B = coordinate position of the center of rectangle B

Bx = unit vector representing the local x-axis of B
By = unit vector representing the local y-axis of B
W_B = half width of B (corresponds with the local x-axis of B)
H_B = half height of B (corresponds with the local y-axis of B)

Then we will have rectangles A and B visually depicted as follows:

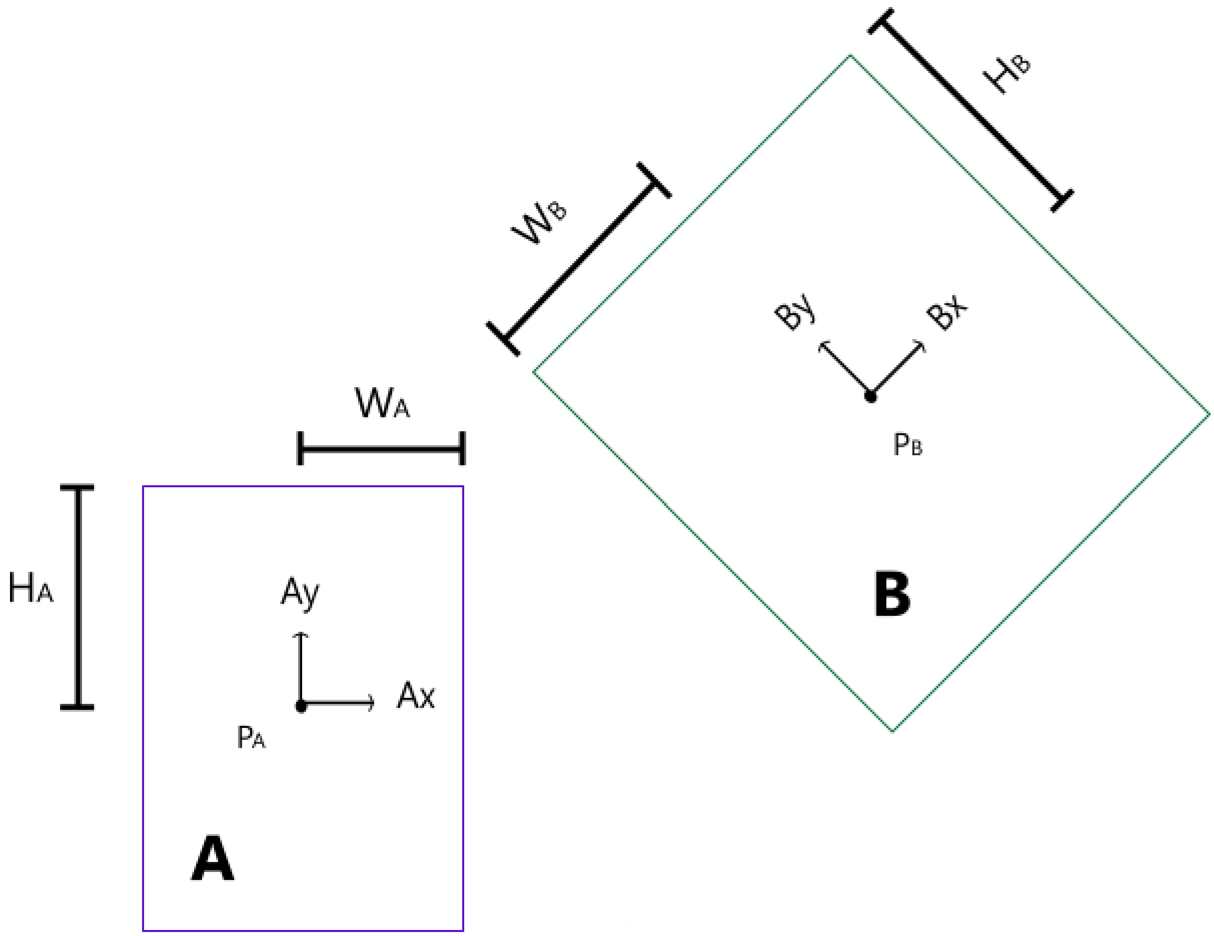


Illustration: Rectangle A and rectangle B with their properties depicted.

Recall that rectangle A and rectangle B does not intersect if and only if a separating axis exists. Then rectangle A and rectangle B intersects if and only if there is no separating axis. Hence, if we find a separating axis, then rectangle A and rectangle B are disjoint and do not intersect. However, if we do not find any separating axes, then rectangle A and rectangle B intersect.

Recall there are 4 possible separating axes for 2 rectangles, and they are perpendicular to the edges of rectangles A and B. However, the edges are parallel to the local XY axes of the rectangles, and the local x-axis and y-axis of each rectangle is perpendicular to one another (e.g.

Ax is perpendicular to Ay). Thus, the 4 possible separating axes we need to check for is Ax, Ay, Bx, and By.

To reduce the complexity and increase the efficiency of the computation, we will not project the entire rectangle A and rectangle B onto each possible separating axis. Instead, we will only project half of each rectangle. Notice we can only do this because we have the mid-point of each rectangle, and the dimensions of each rectangle extend symmetrically on both of its side when a line intersects it in the center.

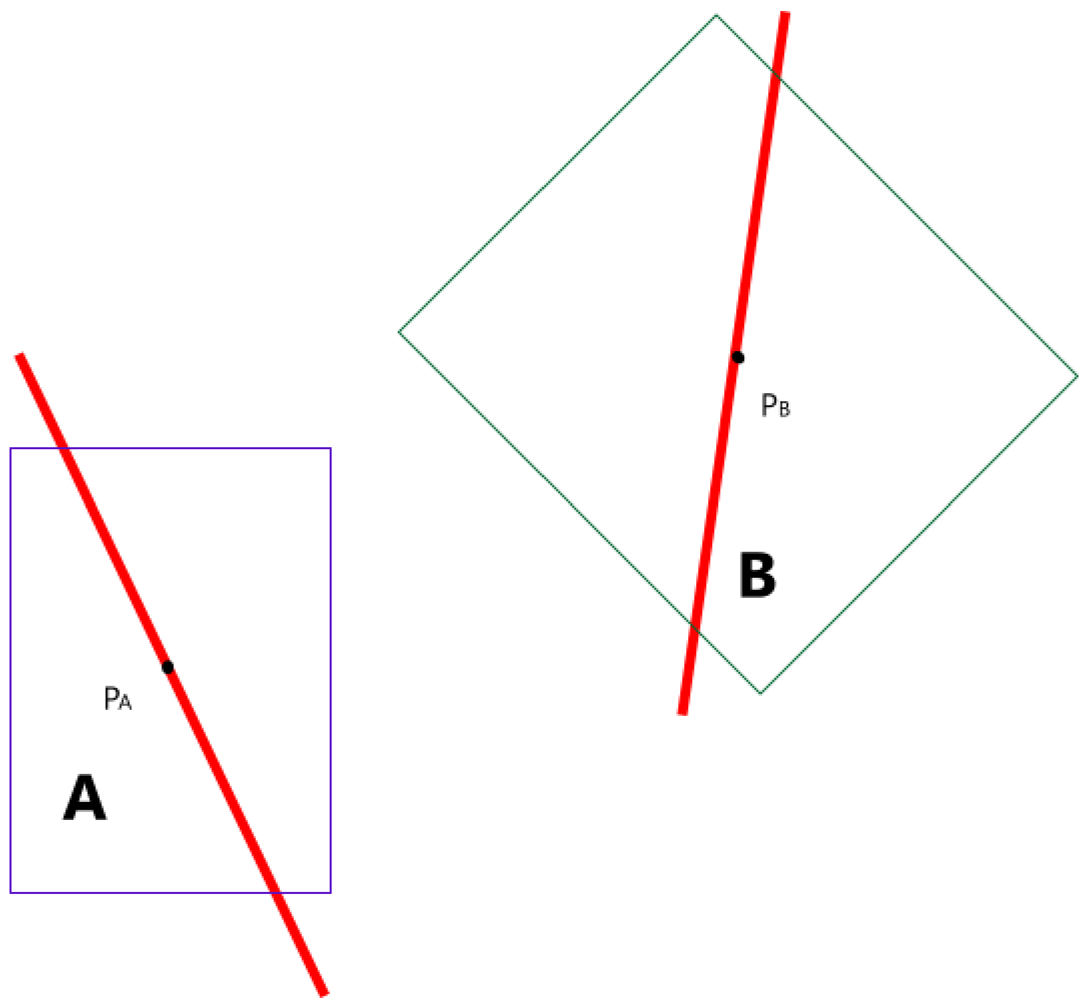


Illustration: No matter how a line divides a rectangle, if the line crosses the center of the rectangle, then the line will bisect the rectangle such that the dimension of the rectangle extends equally on both sides.

Let
$T = P_B - P_A$
L = an arbitrary axis (i.e. a unit vector)
Proj (v) = v • L = projection of v onto an axis, L

Then a separating axis, L, exists if and only if

|Proj (T)| > ½|Proj (RectangleA)| + ½|Proj (RectangleB)|

which is equivalent to

$|Proj (T)| > |Proj (W_A*Ax)| + |Proj (H_A*Ay)| + |Proj (W_B*Bx)| + |Proj(H_B*By)|$

which is equivalent to

$| T \bullet L | > | (W_A*Ax) \bullet L | + | (H_A*Ay) \bullet L | + | (W_B*Bx) \bullet L | + |(H_B*By) \bullet L |$

where | s | denotes the absolute value of scalar s.

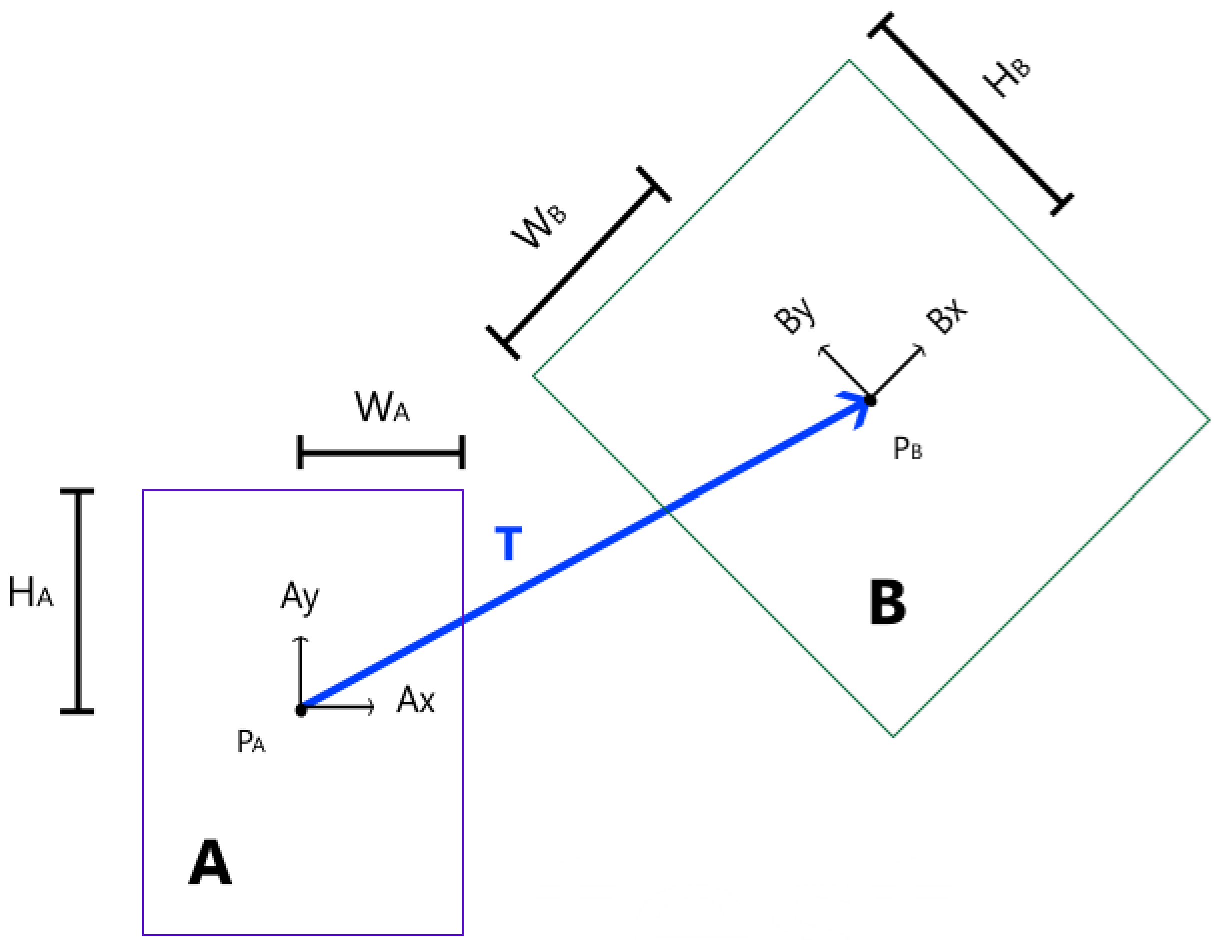


Illustration: T extends from the center of A to the center of B. This is why we only use half the dimensions of rectangle A and rectangle B.

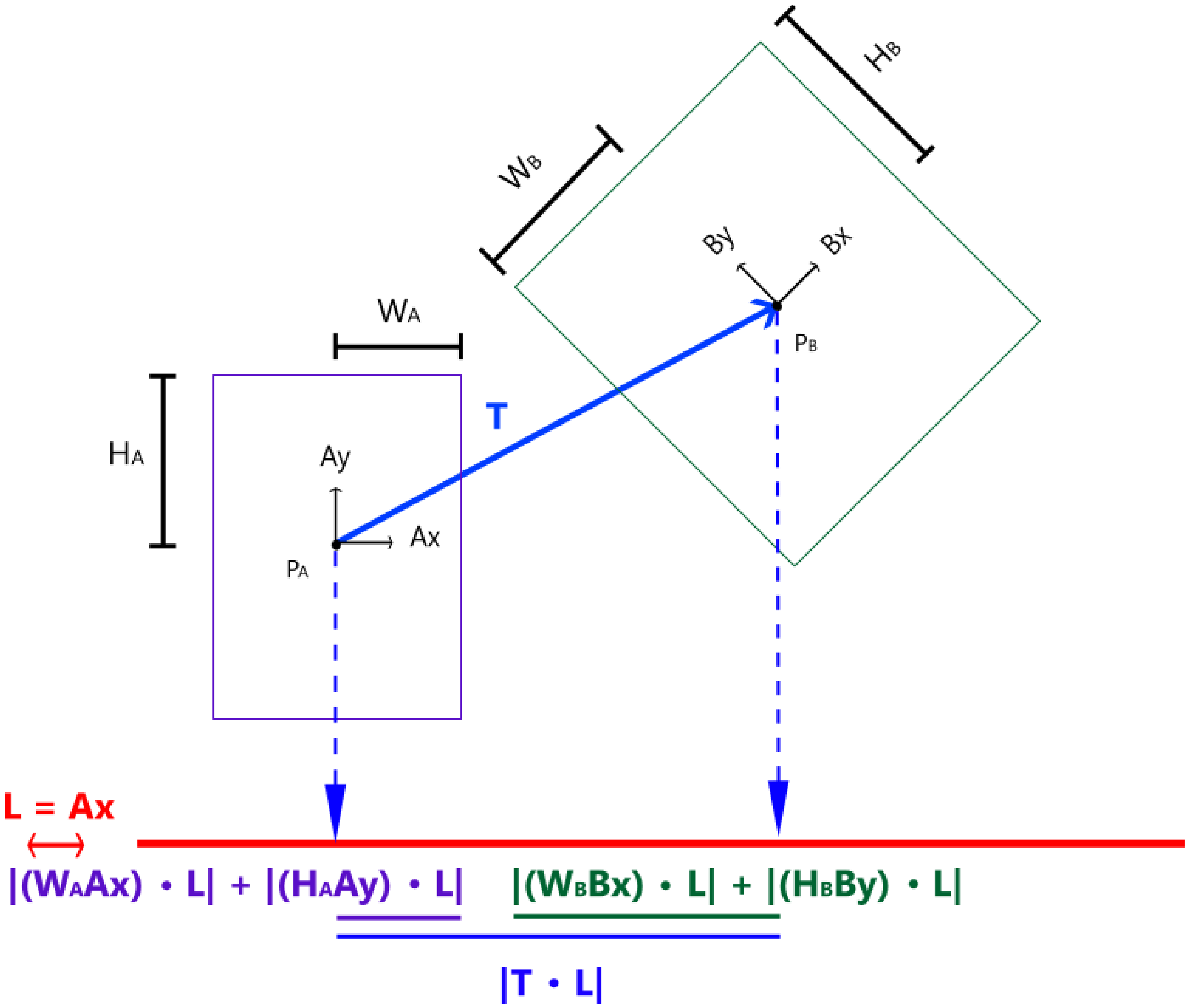


Illustration: The vector T and half of rectangles A and B are projected onto the axis L= Ax (or –Ax). The magnitudes of the projections are compared side by side at the bottom of this illustration. Notice, in this case, $|T \bullet L| > |(W_A Ax) \bullet L| + |(H_A Ay) \bullet L| + |(W_B Bx) \bullet L| + |(H_B By) \bullet L|$. Thus, L = Ax is a separating axis. However, notice that L = Ay and L = By are not separating axes for the case in this illustration.

Here is the general inequality again:

$| T \bullet L | > | (W_A{*}Ax) \bullet L | + | (H_A{*}Ay) \bullet L | + | (W_B{*}Bx) \bullet L | + |(H_B{*}By) \bullet L |$

Recall we have 4 cases to check in 2D space for determining the intersection between two rectangles. Rectangles A and B intersect or collide if and only if the following 4 cases are all false:

CASE 1:
// L = Ax

// If true, L is a separating axis parallel to Ax (and there exists a separating line parallel to Ay)

$| T \bullet Ax | > | (W_A*Ax) \bullet Ax | + | (H_A*Ay) \bullet Ax | + | (W_B*Bx) \bullet Ax | + |(H_B*By) \bullet Ax |$
$| T \bullet Ax | > W_A + 0 + | (W_B*Bx) \bullet Ax | + |(H_B*By) \bullet Ax |$
$| T \bullet Ax | > W_A + | (W_B*Bx) \bullet Ax | + |(H_B*By) \bullet Ax |$

CASE 2:
// L = Ay
// If true, L is a separating axis parallel to Ay (and there exists a separating line parallel to Ax)

$| T \bullet Ay | > H_A + | (W_B*Bx) \bullet Ay | + |(H_B*By) \bullet Ay |$

CASE 3:
// L = Bx
// If true, L is a separating axis parallel to Bx (and there exists a separating line parallel to By)

$| T \bullet Bx | > | (W_A* Ax) \bullet Bx | + | (H_A*Ay) \bullet Bx | + W_B$

CASE 4:
// L = By
// If true, L is a separating axis parallel to By (and there exists a separating line parallel to Bx)

$| T \bullet By | > | (W_A* Ax) \bullet By | + | (H_A*Ay) \bullet By | + H_B$

Separating Axis Theorem and Boxes in 3D Space

Now we move onto the 3D space, and we will show how to compute the intersection or collision between two oriented bounding boxes.

The logic is similar to the one in 2D space, but now instead of only having to consider 4 cases, we need to consider 15 cases of possible separating axes (L).

In 3D space, we no longer have corresponding separating lines that are perpendicular to the separating axes. Instead, we have separating planes that separate the bounding volumes (and they are perpendicular to their corresponding separating axes).

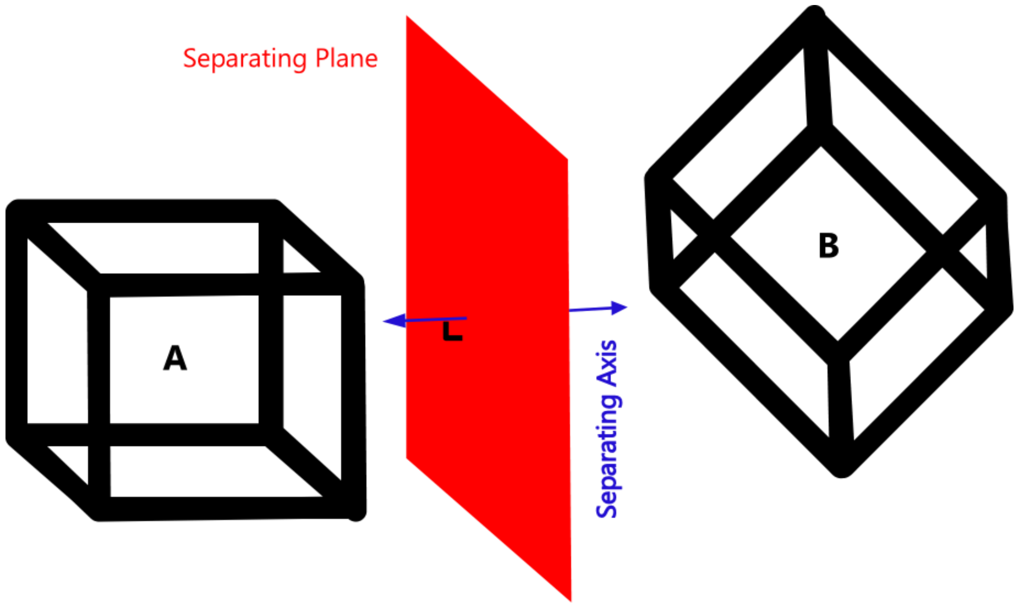


Illustration: The separating plane in 3D space contrasts with the separating line in 2D space. The separating plane separates box A from box B. In this particular illustration, the separating plane is parallel to the right and left faces of box A.

Recall in 2D space, each rectangle has 2 unique axes extended by its edges, and the separating lines (we are interested in) are parallel to these edges. Recall the reason we are only interested in separating lines parallel to one of the edges is because if any separating line is to exist, it has to be parallel to one of the edges.

In 3D space, each box only has 3 unique planes extended by its faces, and the separating planes (we are interested in) are parallel to these faces. We are interested in the separating planes parallel to the faces, but in 3D space, the faces are not the only concern. We are also interested in the edges, which we will discuss later.

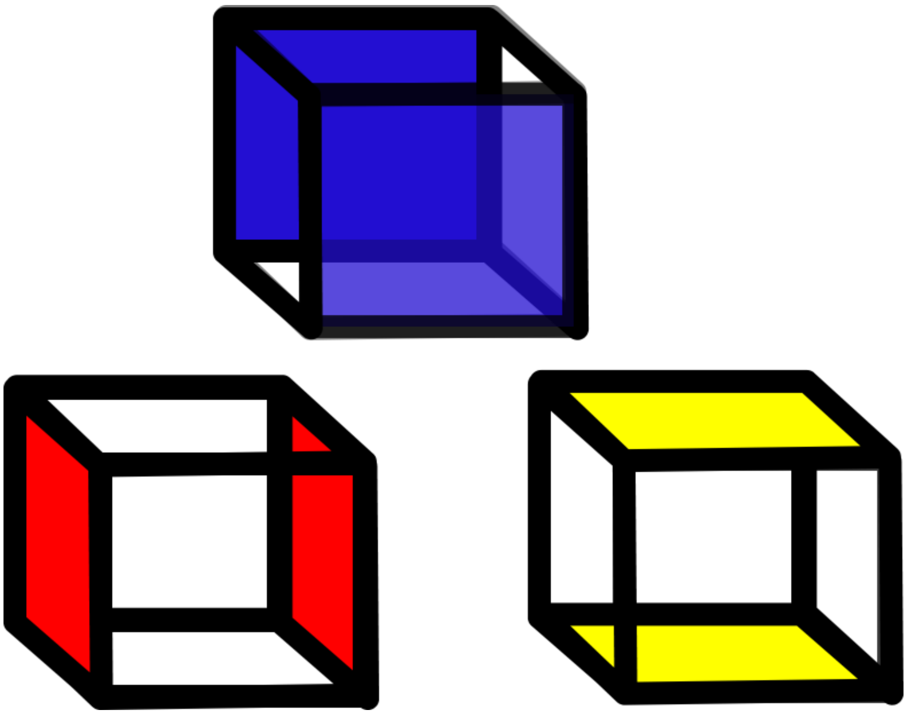


Illustration: The 3 unique planes of a box are parallel to the above 6 faces of the box. Notice a box only has 3 unique planes because each face of a box is parallel to its opposite face. The top-most box highlights the front and back faces as parallel planes. The left-most box highlights the right and left faces as parallel planes, and the right-most box highlights the top and bottom faces as parallel planes.

Recall in 2D space, the separating lines of interest are parallel to the edges of the rectangles, and the separating axes of interest are perpendicular to the separating lines (such that the separating axes of interest are perpendicular to the edges of the rectangles).

Similarly, in 3D space, the separating planes of interest are parallel to the faces of the boxes, and the separating axes of interest are perpendicular to the separating planes. Hence the separating axes of interest are perpendicular to the 3 unique faces of each box. Notice these 6 separating axes of interest correspond to the 6 local (XYZ) axes of the two boxes.

Therefore, there are at least 6 possible separating axes we need to verify.

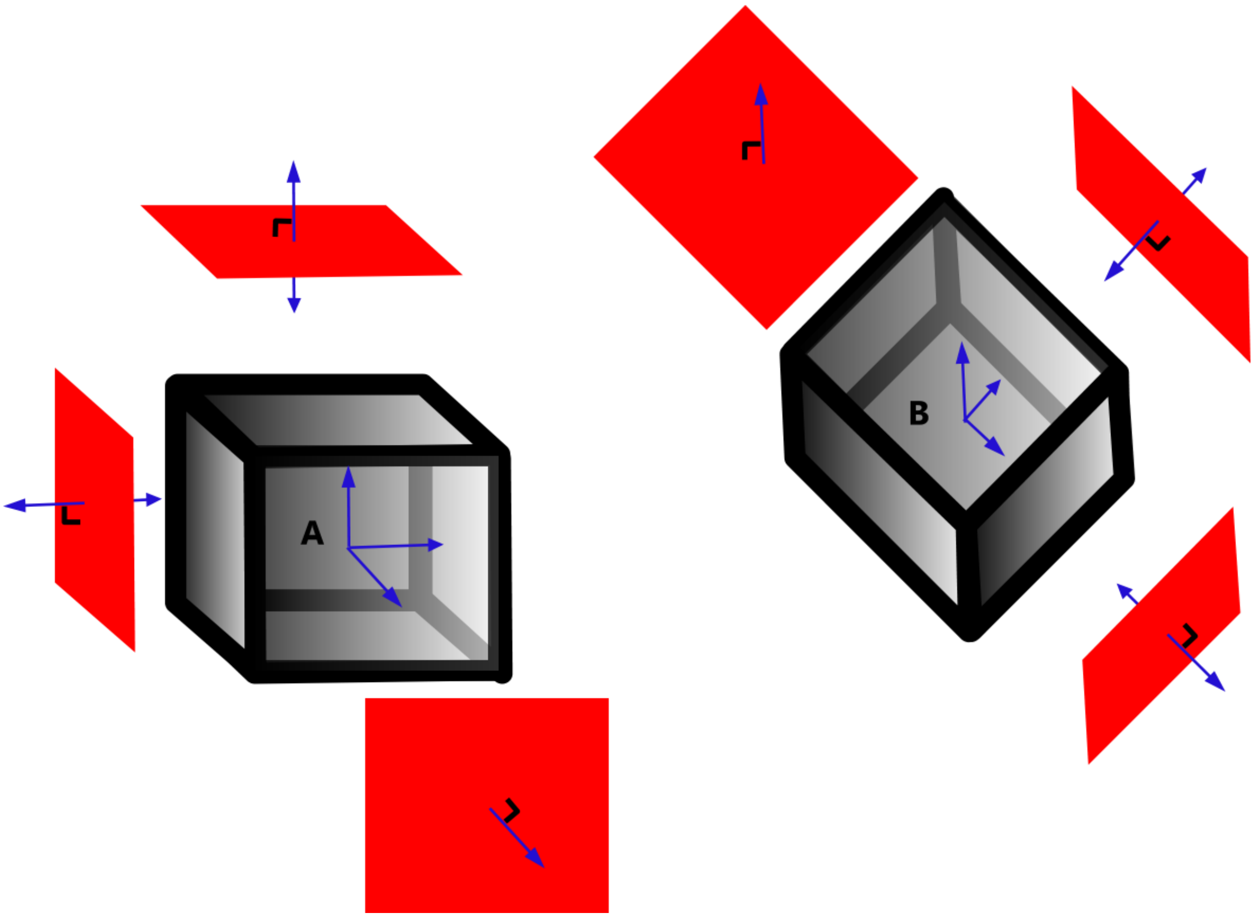
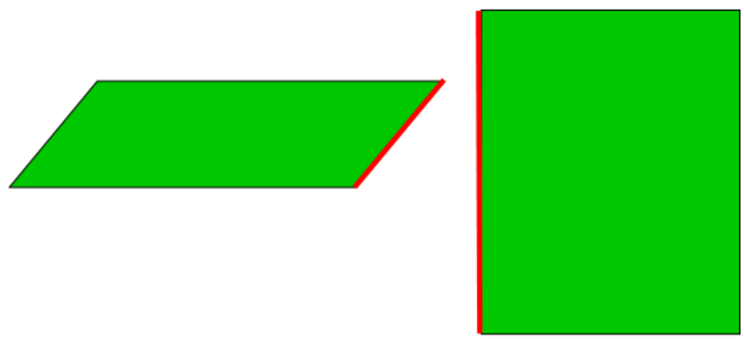


Illustration: The separating planes of interest are depicted in red, while the separating axes of interest are depicted as blue arrows intersecting the red planes. Notice the red separating planes of interest are parallel to the faces of the boxes. Also note the blue separating axes of interest are parallel to the local XYZ axes (also depicted in blue) of the boxes.

In 2D space, we only have the possibility of the edges colliding. In 3D space, we have the possibility of the faces colliding in addition to the edges colliding.

Consider two faces (depicted green) in the illustration below. If we only consider separating planes parallel to the faces, we will detect a collision when there is no collision. There is no separating plane parallel to any of the faces, yet, a separating plane must exists because clearly, the two faces are not intersecting. It turns out there exists a separating plane spanned by the axes of two edges, one from each face.



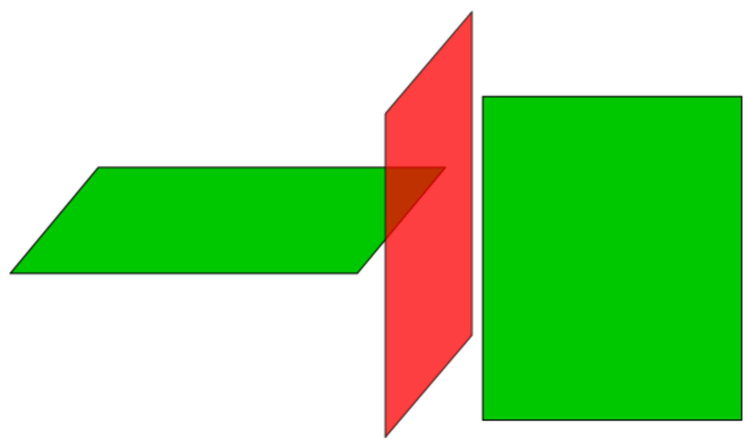


Illustration: The above image depicts two non-intersecting green faces. A separating plane that separates the two faces is spanned by the axes of the two edges highlighted in red. The bottom image depicts a separating plane.

We have already found the 6 possible separating axes corresponding to collisions of the faces of the boxes. Now we need to find the possible separating axes for the edges of the boxes.

Consider two edges (in general) in 3D space (depicted in the illustration below). The two edges collide if and only if there is no separating plane that separates them. However, if any separating plane is to exist between the two edges, there must exists a separating plane that is parallel to the plane spanned by the axes of the two edges.

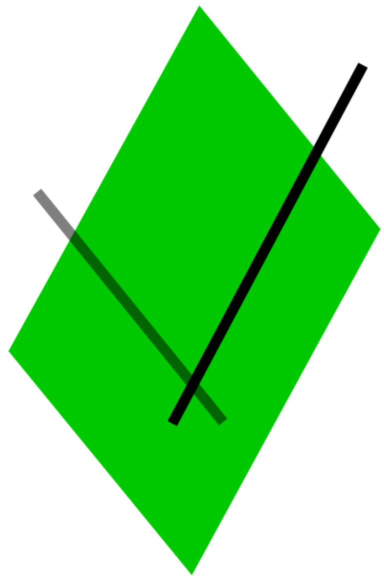
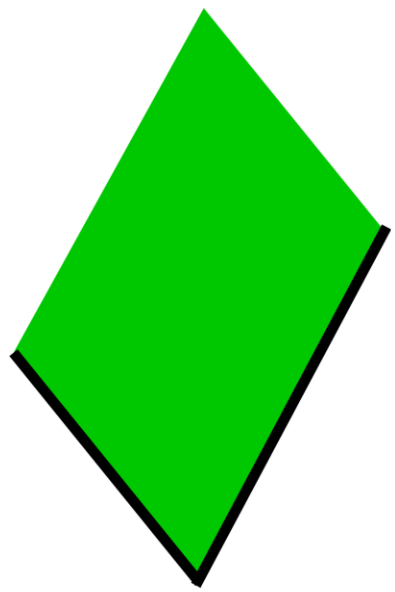
 

Illustration: In 3D space, the picture on the left depicts a green separating plane between the two edges. Since the plane exists and separates the two edges, it must be the case that the two edges do not intersect. The picture on the right depicts the separating plane parallel to the plane spanned by the axes of the two edges. Note: The separating plane spans infinitely and is not finite as depicted here in the illustration.

Notice there are 3 unique axes extended by the edges of each box (corresponding to the local XYZ axes of the box).

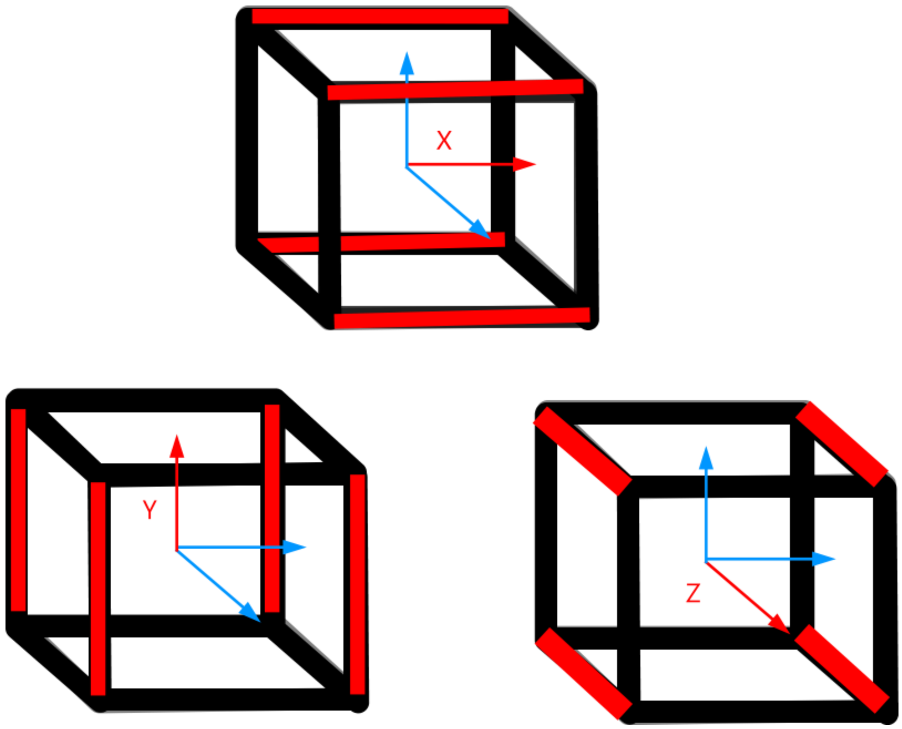


Illustration: The 12 edges of a box and their corresponding parallel axes. Notice for the top-most box, there are 4 edges parallel to the local x-axis of the box.

Since the separating plane is spanned by the combination of an edge from one box and an edge from the other box, there are 3*3 = 9 such possible combinations between the 3 local axes from each box for the 2 boxes. Then there are 9 possible separating planes just for considering these edges. (That is, the 9 combinations are (Ax, Bx), (Ax, By), (Ax, Bz), (Ay, Bx), (Ay, By), (Ay, Bz), (Az, Bx), (Az, By), (Az, Bz).)

Again, the possible separating axes are perpendicular to the separating planes. Since each separating plane is spanned by an edge from one box and an edge from the other box, we can take the cross product of the axes of these two edges to find the corresponding perpendicular separating axes.

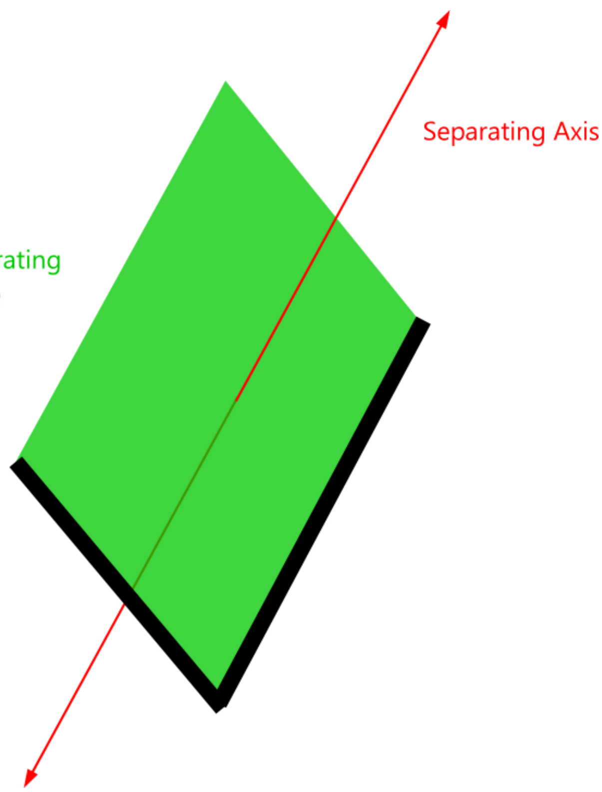


Illustration: The separating axis perpendicular to the separating plane can be found by taking the cross product of the axes of the two edges. The two edges are the black lines in the illustration.

Since the edges of each box are parallel to the box's local axes, we just take the cross products between the local (XYZ) axes of the two boxes to find the 9 possible separating axes.

So there are 9 separating axes to consider for edges collision in addition to the 6 separating axes we already have found for the faces collision. This makes the total number of possible separating axes to consider at 15. This is substantially more compared to the 4 possible separating axes to consider in 2D space for 2 rectangles.

Case Studies: Detecting Intersecting Boxes

Here are the 15 possible separating axes (L).

CASE 1:
L = Ax

CASE 2:
L = Ay

CASE 3:

$L = Az$

CASE 4:
$L = Bx$

CASE 5:
$L = By$

CASE 6:
$L = Bz$

CASE 7:
$L = Ax \times Bx$

CASE 8:
$L = Ax \times By$

CASE 9:
$L = Ax \times Bz$

CASE 10:
$L = Ay \times Bx$

CASE 11:
$L = Ay \times By$

CASE 12:
$L = Ay \times Bz$

CASE 13:
$L = Az \times Bx$

CASE 14:
$L = Az \times By$

CASE 15:
$L = Az \times Bz$

We now present the projection of a box onto an axis, and this is not much different than the projection of a rectangle onto an axis. The only difference is that the box has to accommodate an additional dimension.

Consider a box A and an axis L.

Suppose box A has the following properties:

P_A = coordinate position of the center of A
Ax = unit vector representing the x-axis of A
Ay = unit vector representing the y-axis of A
Az = unit vector representing the z-axis of A
W_A = half width of A (corresponds with the local x-axis of A)
H_A = half height of A (corresponds with the local y-axis of A)
D_A = half depth of A (corresponds with the local z-axis of A)

The projection of half of a box A onto an axis L is given by the equation

$$\begin{aligned} \tfrac{1}{2}|\text{Proj}(\text{BoxA})| &= |\text{Proj}(W_A{*}Ax)| + |\text{Proj}(H_A{*}Ay)| + |\text{Proj}(D_A{*}Az)| \\ &= |(W_A{*}Ax) \bullet L| + |(H_A{*}Ay) \bullet L| + |(D_A{*}Az) \bullet L| \end{aligned}$$

Consider another box B with the following properties:

P_B = coordinate position of the center of B
Bx = unit vector representing the x-axis of B
By = unit vector representing the y-axis of B
Bz = unit vector representing the z-axis of B
W_B = half width of B (corresponds with the local x-axis of B)
H_B = half height of B (corresponds with the local y-axis of B)
D_B = half depth of B (corresponds with the local z-axis of B)

Let

$T = P_B - P_A$

Then the inequality we want to use to check for a collision is:

$$\begin{aligned} |\text{Proj}(T)| > {} & |\text{Proj}(W_A{*}Ax)| + |\text{Proj}(H_A{*}Ay)| + |\text{Proj}(D_A{*}Az)| \\ & + |\text{Proj}(W_B{*}Bx)| + |\text{Proj}(H_B{*}By)| + |\text{Proj}(D_B{*}Bz)| \end{aligned}$$

which is equivalent to

$$\begin{aligned} |T \bullet L| > {} & |(W_A{*}Ax) \bullet L| + |(H_A{*}Ay) \bullet L| + |(D_A{*}Az) \bullet L| \\ & + |(W_B{*}Bx) \bullet L| + |(H_B{*}By) \bullet L| + |(D_B{*}Bz) \bullet L| \end{aligned}$$

where | s | denotes the absolute value of scalar s.

Computing the Intersection between Two Rectangles in 3D Space is Problematic

Caution:

While I mentioned 2D faces in 3D space, our separating axis theorem approach does not exactly work for collisions between two 2D objects in 3D space (unless the two rectangles are actually boxes, such that each box has a dimension equal to zero).

Consider two 2D faces side-by-side lying in the same plane. No separating axis will be found by our algorithm in 3D space even though the separating axis does exist. This is because the only separating plane of interest is in between. Unfortunately, each rectangle only has one orthogonal axis, and the span of the two orthogonal axes of the rectangles does not create a plane because the two orthogonal axes are the same.

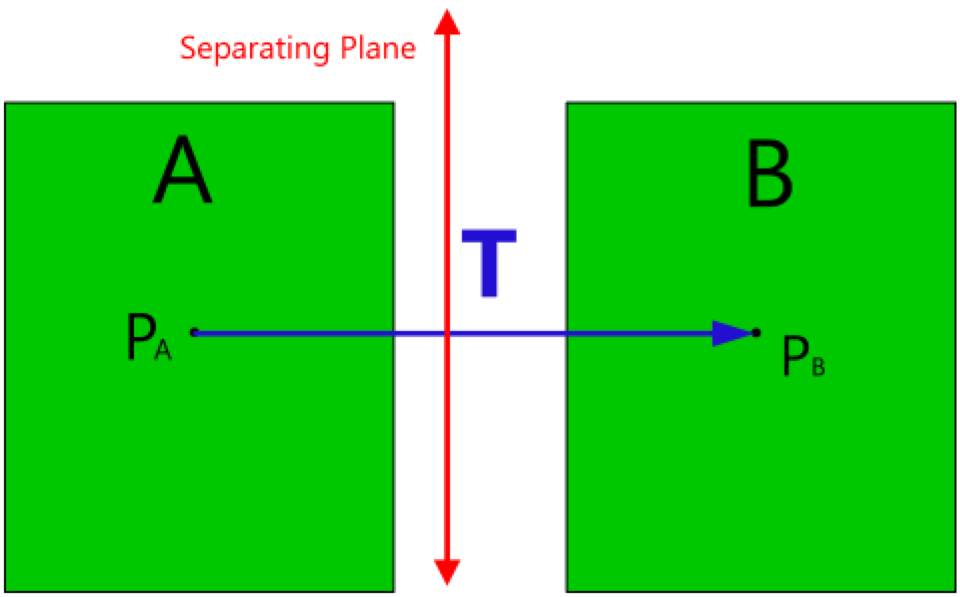


Illustration: Two 2D faces side-by-side lying in the same plane in 3D space. Clearly the two faces are not intersecting, but our approach in 3D space will interpret the two faces as intersecting. Notice, in this case, the only separating axes checked are perpendicular to both of the faces. Also, notice these separating axes checked cannot be depicted in the illustration above because they are going in and coming out of the page.

Since the separating axes checked are perpendicular to the corresponding separating plane, the vector, $T = P_B - P_A$, projected onto these separating axes will always be 0. The projection of the two faces onto these separating axes will also be 0. However, T = 0 is never greater than 0. Hence, the two 2D faces will always be interpreted as colliding even though they are not intersecting one another.

Note, however, that our approach will interpret the collision between one 2D object and one 3D object in 3D space correctly. There is only a problem when we consider the intersection between two 2D objects in 3D space, and the problem should only occurs when the two faces are coplanar (existing in the same plane), similar to the illustration above. It is still possible to detect the collision between two 2D objects in 3D space by considering a combination of separating lines (in 2D space) and separating planes (in 3D space).

Computing the Intersection of Two Oriented Bounding Boxes

Let A and B be oriented bounding boxes (OBB).

Parameters

P_A = coordinate position of the center of A
Ax = unit vector representing the x-axis of A
Ay = unit vector representing the y-axis of A
Az = unit vector representing the z-axis of A
W_A = half width of A (corresponds with the local x-axis of A)
H_A = half height of A (corresponds with the local y-axis of A)
D_A = half depth of A (corresponds with the local z-axis of A)

P_B = coordinate position of the center of B
Bx = unit vector representing the x-axis of B
By = unit vector representing the y-axis of B
Bz = unit vector representing the z-axis of B
W_B = half width of B (corresponds with the local x-axis of B)
H_B = half height of B (corresponds with the local y-axis of B)
D_B = half depth of B (corresponds with the local z-axis of B)

Variables

$T = P_B - P_A$
$R_{ij} = A_i \bullet B_j$

Here is the general inequality:

$$|T \bullet L| > |W_A Ax \bullet L| + |H_A Ay \bullet L| + |D_A Az \bullet L| + |W_B Bx \bullet L| + |H_B By \bullet L| + |D_B Bz \bullet L|$$

Here are the 15 separating axes we need to inspect for 2 boxes in 3D space:

CASE 1:
$// L = Ax$

$$\left|T \bullet Ax\right| > |W_A Ax \bullet Ax| + |H_A Ay \bullet Ax| + |D_A Az \bullet Ax| + |W_B Bx \bullet Ax| + |H_B By \bullet Ax| + |D_B Bz \bullet Ax|$$
$$\left|T \bullet Ax\right| > |W_A| + 0 + 0 + |W_B Bx \bullet Ax| + |H_B By \bullet Ax| + |D_B Bz \bullet Ax|$$
$$\left|T \bullet Ax\right| > W_A + |W_B Rxx| + |H_B Rxy| + |D_B Rxz|$$

CASE 2:
$// L = Ay$

$$\left|T \bullet Ay\right| > |W_A Ax \bullet Ay| + |H_A Ay \bullet Ay| + |D_A Az \bullet Ay| + |W_B Bx \bullet Ay| + |H_B By \bullet Ay| + |D_B Bz \bullet Ay|$$
$$\left|T \bullet Ay\right| > 0 + |H_A| + 0 + |W_B Bx \bullet Ay| + |H_B By \bullet Ay| + |D_B Bz \bullet Ay|$$
$$\left|T \bullet Ay\right| > H_A + |W_B Ryx| + |H_B Ryy| + |D_B Ryz|$$

CASE 3:
$// L = Az$

$$\left|T \bullet Az\right| > |W_A Ax \bullet Az| + |H_A Ay \bullet Az| + |D_A Az \bullet Az| + |W_B Bx \bullet Az| + |H_B By \bullet Az| + |D_B Bz \bullet Az|$$
$$\left|T \bullet Az\right| > 0 + 0 + |D_A| + |W_B Bx \bullet Az| + |H_B By \bullet Az| + |D_B Bz \bullet Az|$$
$$\left|T \bullet Az\right| > D_A + |W_B Rzx| + |H_B Rzy| + |D_B Rzz|$$

CASE 4:
$// L = Bx$

$$\left|T \bullet Bx\right| > |W_A Ax \bullet Bx| + |H_A Ay \bullet Bx| + |D_A Az \bullet Bx| + |W_B Bx \bullet Bx| + |H_B By \bullet Bx| + |D_B Bz \bullet Bx|$$
$$\left|T \bullet Bx\right| > |W_A Ax \bullet Bx| + |H_A Ay \bullet Bx| + |D_A Az \bullet Bx| + |W_B| + 0 + 0$$
$$\left|T \bullet Bx\right| > |W_A Rxx| + |H_A Ryx| + |D_A Rzx| + W_B$$

CASE 5:
$// L = By$

$$\left|T \bullet By\right| > |W_A Ax \bullet By| + |H_A Ay \bullet By| + |D_A Az \bullet By| + |W_B Bx \bullet By| + |H_B By \bullet By| + |D_B Bz \bullet By|$$
$$\left|T \bullet By\right| > |W_A Ax \bullet By| + |H_A Ay \bullet By| + |D_A Az \bullet By| + 0 + |H_B| + 0$$
$$\left|T \bullet By\right| > |W_A Rxy| + |H_A Ryy| + |D_A Rzy| + H_B$$

CASE 6:
$// L = Bz$

$$\left|T \bullet Bz\right| > |W_A Ax \bullet Bz| + |H_A Ay \bullet Bz| + |D_A Az \bullet Bz| + |W_B Bx \bullet Bz| + |H_B By \bullet Bz| + |D_B Bz \bullet Bz|$$
$$\left|T \bullet Bz\right| > |W_A Ax \bullet Bz| + |H_A Ay \bullet Bz| + |D_A Az \bullet Bz| + 0 + 0 + |D_B|$$
$$\left|T \bullet Bz\right| > |W_A Rxz| + |H_A Ryz| + |D_A Rzz| + D_B$$

CASE 7:
$// L = Ax \times Bx$

$$\left|T \bullet (Ax \times Bx)\right| > |W_A Ax \bullet (Ax \times Bx)| + |H_A Ay \bullet (Ax \times Bx)| + |D_A Az \bullet (Ax \times Bx)| + |W_B Bx \bullet (Ax \times Bx)| + |H_B By \bullet (Ax \times Bx)| + |D_B Bz \bullet (Ax \times Bx)|$$

PROOF $sA \bullet (B \times C) = sC \bullet (A \times B)$

$$\left|T \bullet (Ax \times Bx)\right| > |W_A Bx \bullet (Ax \times Ax)| + |H_A Bx \bullet (Ay \times Ax)| + |D_A Bx \bullet (Az \times Ax)| + |W_B Bx \bullet -(Bx \times Ax)| + |H_B By \bullet -(Bx \times Ax)| + |D_B Bz \bullet -(Bx \times Ax)|$$
$$\left|T \bullet (Ax \times Bx)\right| > |W_A Bx \bullet (Ax \times Ax)| + |H_A Bx \bullet (Ay \times Ax)| + |D_A Bx \bullet (Az \times Ax)| + |-W_B Ax \bullet (Bx \times Bx)| + |-H_B Ax \bullet (By \times Bx)| + |-D_B Ax \bullet (Bz \times Bx)|$$
$$\left|T \bullet (Ax \times Bx)\right| > 0 + |H_A Bx \bullet -Az| + |D_A Bx \bullet Ay| + 0 + |-H_B Ax \bullet -Bz| + |-D_B Ax \bullet By|$$
$$\left|T \bullet (Ax \times Bx)\right| > |-H_A Bx \bullet Az| + |D_A Bx \bullet Ay| + |H_B Ax \bullet Bz| + |-D_B Ax \bullet By|$$
$$\left|T \bullet (Ax \times Bx)\right| > |H_A Rzx| + |D_A Ryx| + |H_B Rxz| + |D_B Rxy|$$

Similarly, for the following cases…

CASE 8:
$// L = Ax \times By$

$$\left|T \bullet (Ax \times By)\right| > 0 + |H_A Ay \bullet (Ax \times By)| + |D_A Az \bullet (Ax \times By)| + |W_B Bx \bullet (Ax \times By)| + 0 + |D_B Bz \bullet (Ax \times By)|$$
$$\left|T \bullet (Ax \times By)\right| > |H_A Rzy| + |D_A Ryy| + |W_B Rxz| + |D_B Rxx|$$

CASE 9:
$// L = Ax \times Bz$

$$\left|T \bullet (Ax \times Bz)\right| > 0 + |H_A Ay \bullet (Ax \times Bz)| + |D_A Az \bullet (Ax \times Bz)| + |W_B Bx \bullet (Ax \times Bz)| + |H_B By \bullet (Ax \times Bz)| + 0$$
$$\left|T \bullet (Ax \times Bz)\right| > |H_A Rzz| + |D_A Ryz| + |W_B Rxy| + |H_B Rxx|$$

CASE 10:
$// L = Ay \times Bx$

$$\left|T \bullet (Ay \times Bx)\right| > |W_A Ax \bullet (Ay \times Bx)| + 0 + |D_A Az \bullet (Ay \times Bx)| + 0 + |H_B By \bullet (Ay \times Bx)| + |D_B Bz \bullet (Ay \times Bx)|$$
$$\left|T \bullet (Ay \times Bx)\right| > |W_A Rzx| + |D_A Rxx| + |H_B Ryz| + |D_B Ryy|$$

CASE 11:
$// L = Ay \times By$

$$\left|T \bullet (Ay \times By)\right| > |W_A Ax \bullet (Ay \times By)| + 0 + |D_A Az \bullet (Ay \times By)| + |W_B Bx \bullet (Ay \times By)| + 0 + |D_B Bz \bullet (Ay \times By)|$$
$$\left|T \bullet (Ay \times By)\right| > |W_A Rzy| + |D_A Rxy| + |W_B Ryz| + |D_B Ryx|$$

CASE 12:
$// L = Ay \times Bz$

$$\left|T \bullet (Ay \times Bz)\right| > |W_A Ax \bullet (Ay \times Bz)| + 0 + |D_A Az \bullet (Ay \times Bz)| + |W_B Bx \bullet (Ay \times Bz)| + |H_B By \bullet (Ay \times Bz)| + 0$$
$$\left|T \bullet (Ay \times Bz)\right| > |W_A Rzz| + |D_A Rxz| + |W_B Ryy| + |H_B Ryx|$$

CASE 13:
$// L = Az \times Bx$

$$\left|T \bullet (Az \times Bx)\right| > |W_A Ax \bullet (Az \times Bx)| + |H_A Ay \bullet (Az \times Bx)| + 0 + 0 + |H_B By \bullet (Az \times Bx)| + |D_B Bz \bullet (Az \times Bx)|$$
$$\left|T \bullet (Az \times Bx)\right| > |W_A Ryx| + |H_A Rxx| + |H_B Rzz| + |D_B Rzy|$$

CASE 14:
$// L = Az \times By$

$$\left|T \bullet (Az \times By)\right| > |W_A Ax \bullet (Az \times By)| + |H_A Ay \bullet (Az \times By)| + 0 + |W_B Bx \bullet (Az \times By)| + 0 + |D_B Bz \bullet (Az \times By)|$$
$$\left|T \bullet (Az \times By)\right| > |W_A Ryy| + |H_A Rxy| + |W_B Rzz| + |D_B Rzx|$$

CASE 15:
$// L = Az \times Bz$

$$\left|T \bullet (Az \times Bz)\right| > |W_A Ax \bullet (Az \times Bz)| + |H_A Ay \bullet (Az \times Bz)| + 0 + |W_B Bx \bullet (Az \times Bz)| + |H_B By \bullet (Az \times Bz)| + 0$$
$$\left|T \bullet (Az \times Bz)\right| > |W_A Ryz| + |H_A Rxz| + |W_B Rzy| + |H_B Rzx|$$

Optimizing the Computation of T•L

For those interested in optimizations, we can optimize the computations for T•L, whenever L requires a cross product.

PROOF $T \bullet (Ax \times Bx) = (T \bullet Az)(Ay \bullet Bx) - (T \bullet Ay)(Az \bullet Bx)$

As a result of the above derivation, we can interpret the other 8 possible separating axes by using our knowledge of the XYZ-axes, and the right-hand rule (for cross products).

Example:

The above equation we proved is with respect to the x-axis of A.

For $T \bullet (Ay \times Bx)$, we need the equation in respect to the y-axis of A. Refer to the illustration below. With respect to the y-axis compared to with respect to the x-axis, Y corresponds with X, Z corresponds with Y, and X corresponds with Z. Hence, we replace all Ax with Ay, all Ay with Az, and all Az with Ax in the equation, and we get the following result:

$$T \bullet (Ay \times Bx) = (T \bullet Ax)(Az \bullet Bx) - (T \bullet Az)(Ax \bullet Bx)$$

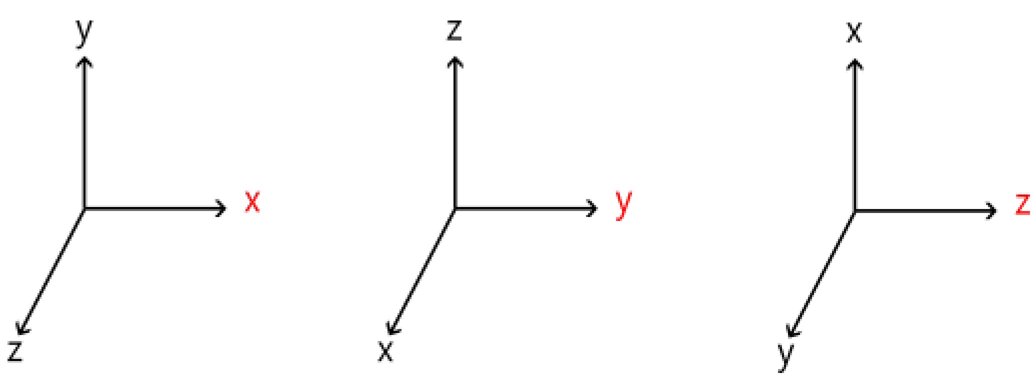


Illustration: We orient the XYZ coordinate system with respect to Y and Z to make their corresponding axes to the XYZ coordinate system with respect to X more apparent. Notice Y in the system with respect to X is equivalent to Z in the system with respect to Y.

The illustration below shows how we derive the XYZ coordinate systems with respect to Y and Z through rotations.

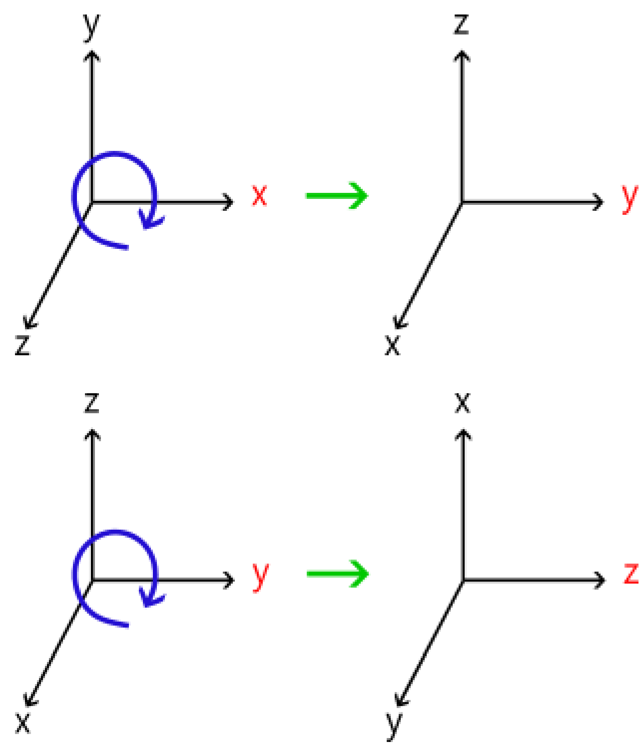


Illustration: 1) From X, we rotate the coordinate system to get the system with respect to Y.
2) From Y, we rotate the coordinate system to get the system with respect to Z.

Here are the optimized T•L equations for finding the 9 possible separating axes (for collisions between edges):

$$// L = Ax \times Bx$$

$$T \bullet (Ax \times Bx) = (T \bullet Az)(Ay \bullet Bx) - (T \bullet Ay)(Az \bullet Bx)$$
$$T \bullet (Ax \times Bx) = (T \bullet Az)Ryx - (T \bullet Ay)Rzx$$

$$// L = Ax \times By$$

$$T \bullet (Ax \times By) = (T \bullet Az)(Ay \bullet By) - (T \bullet Ay)(Az \bullet By)$$
$$T \bullet (Ax \times By) = (T \bullet Az)Ryy - (T \bullet Ay)Rzy$$

$$// L = Ax \times Bz$$

$$T \bullet (Ax \times Bz) = (T \bullet Az)(Ay \bullet Bz) - (T \bullet Ay)(Az \bullet Bz)$$
$$T \bullet (Ax \times Bz) = (T \bullet Az)Ryz - (T \bullet Ay)Rzz$$

$// L = Ay \times Bx$

$$T \bullet (Ay \times Bx) = (T \bullet Ax)(Az \bullet Bx) - (T \bullet Az)(Ax \bullet Bx)$$
$$T \bullet (Ay \times Bx) = (T \bullet Ax)Rzx - (T \bullet Az)Rxx$$

$// L = Ay \times By$

$$T \bullet (Ay \times By) = (T \bullet Ax)(Az \bullet By) - (T \bullet Az)(Ax \bullet By)$$
$$T \bullet (Ay \times By) = (T \bullet Ax)Rzy - (T \bullet Az)Rxy$$

$// L = Ay \times Bz$

$$T \bullet (Ay \times Bz) = (T \bullet Ax)(Az \bullet Bz) - (T \bullet Az)(Ax \bullet Bz)$$
$$T \bullet (Ay \times Bz) = (T \bullet Ax)Rzz - (T \bullet Az)Rxz$$

$// L = Az \times Bx$

$$T \bullet (Az \times Bx) = (T \bullet Ay)(Ax \bullet Bx) - (T \bullet Ax)(Ay \bullet Bx)$$
$$T \bullet (Az \times Bx) = (T \bullet Ay)Rxx - (T \bullet Ax)Ryx$$

$// L = Az \times By$

$$T \bullet (Az \times By) = (T \bullet Ay)(Ax \bullet By) - (T \bullet Ax)(Ay \bullet By)$$
$$T \bullet (Az \times By) = (T \bullet Ay)Rxy - (T \bullet Ax)Ryy$$

$// L = Az \times Bz$

$$T \bullet (Az \times Bz) = (T \bullet Ay)(Ax \bullet Bz) - (T \bullet Ax)(Ay \bullet Bz)$$
$$T \bullet (Az \times Bz) = (T \bullet Ay)Rxz - (T \bullet Ax)Ryz$$

Optimized Computation of OBBs Intersections

After our optimization of T•L, again, we present the 15 separating axis inequalities we need to inspect to two oriented bounding boxes in 3D space.

Recall if at least one of these 15 inequalities is true, then there exists a separating axis. If a separating axis exists, then the two OBBs (A and B) do not intersect, and there is no need to further check the other cases. The two OBBs intersect if and only if all 15 inequalities are false.

For reference:

Let A and B be oriented bounding boxes (OBB).

Parameters

P_A = coordinate position of the center of A
Ax = unit vector representing the x-axis of A
Ay = unit vector representing the y-axis of A
Az = unit vector representing the z-axis of A
W_A = half width of A (corresponds with the local x-axis of A)
H_A = half height of A (corresponds with the local y-axis of A)
D_A = half depth of A (corresponds with the local z-axis of A)

P_B = coordinate position of the center of B
Bx = unit vector representing the x-axis of B
By = unit vector representing the y-axis of B
Bz = unit vector representing the z-axis of B
W_B = half width of B (corresponds with the local x-axis of B)
H_B = half height of B (corresponds with the local y-axis of B)
D_B = half depth of B (corresponds with the local z-axis of B)

Variables

$T = P_B - P_A$
$R_{ij} = A_i \bullet B_j$

Note: | c | = absolute value of c

Cases

CASE 1:
$// L = Ax$
// If true, L is a separating axis parallel to Ax (and there exists a separating plane with normal, Ax)

$$|T \bullet Ax| > W_A + |W_B Rxx| + |H_B Rxy| + |D_B Rxz|$$

CASE 2:
$// L = Ay$
// If true, L is a separating axis parallel to Ay (and there exists a separating plane with normal, Ay)

$$|T \bullet Ay| > H_A + |W_B Ryx| + |H_B Ryy| + |D_B Ryz|$$

CASE 3:
$// L = Az$
// If true, L is a separating axis parallel to Az (and there exists a separating plane with normal, Az)

$$|T \bullet Az| > D_A + |W_B Rzx| + |H_B Rzy| + |D_B Rzz|$$

CASE 4:
$// L = Bx$
// If true, L is a separating axis parallel to Bx (and there exists a separating plane with normal, Bx)

$$|T \bullet Bx| > |W_A Rxx| + |H_A Ryx| + |D_A Rzx| + W_B$$

CASE 5:
$// L = By$
// If true, L is a separating axis parallel to By (and there exists a separating plane with normal, By)

$$|T \bullet By| > |W_A Rxy| + |H_A Ryy| + |D_A Rzy| + H_B$$

CASE 6:
$// L = Bz$
// If true, L is a separating axis parallel to Bz (and there exists a separating plane with normal, Bz)

$$|T \bullet Bz| > |W_A Rxz| + |H_A Ryz| + |D_A Rzz| + D_B$$

CASE 7:
$// L = Ax \times Bx$
// If true, L is a separating axis perpendicular to the separating plane spanned by Ax and Bx

$$|(T \bullet Az)Ryx - (T \bullet Ay)Rzx| > |H_A Rzx| + |D_A Ryx| + |H_B Rxz| + |D_B Rxy|$$

CASE 8:
$// L = Ax \times By$
// If true, L is a separating axis perpendicular to the separating plane spanned by Ax and By

$$|(T \bullet Az)Ryy - (T \bullet Ay)Rzy| > |H_A Rzy| + |D_A Ryy| + |W_B Rxz| + |D_B Rxx|$$

CASE 9:

$// L = Ax \times Bz$

// If true, L is a separating axis perpendicular to the separating plane spanned by Ax and Bz

$$\left|(T \bullet Az)Ryz - (T \bullet Ay)Rzz\right| > |H_A Rzz| + |D_A Ryz| + |W_B Rxy| + |H_B Rxx|$$

CASE 10:

$// L = Ay \times Bx$

// If true, L is a separating axis perpendicular to the separating plane spanned by Ay and Bx

$$\left|(T \bullet Ax)Rzx - (T \bullet Az)Rxx\right| > |W_A Rzx| + |D_A Rxx| + |H_B Ryz| + |D_B Ryy|$$

CASE 11:

$// L = Ay \times By$

// If true, L is a separating axis perpendicular to the separating plane spanned by Ay and By

$$\left|(T \bullet Ax)Rzy - (T \bullet Az)Rxy\right| > |W_A Rzy| + |D_A Rxy| + |W_B Ryz| + |D_B Ryx|$$

CASE 12:

$// L = Ay \times Bz$

// If true, L is a separating axis perpendicular to the separating plane spanned by Ay and Bz

$$\left|(T \bullet Ax)Rzz - (T \bullet Az)Rxz\right| > |W_A Rzz| + |D_A Rxz| + |W_B Ryy| + |H_B Ryx|$$

CASE 13:

$// L = Az \times Bx$

// If true, L is a separating axis perpendicular to the separating plane spanned by Az and Bx

$$\left|(T \bullet Ay)Rxx - (T \bullet Ax)Ryx\right| > |W_A Ryx| + |H_A Rxx| + |H_B Rzz| + |D_B Rzy|$$

CASE 14:

$// L = Az \times By$

// If true, L is a separating axis perpendicular to the separating plane spanned by Az and By

$$\left|(T \bullet Ay)Rxy - (T \bullet Ax)Ryy\right| > |W_A Ryy| + |H_A Rxy| + |W_B Rzz| + |D_B Rzx|$$

CASE 15:

$// L = Az \times Bz$

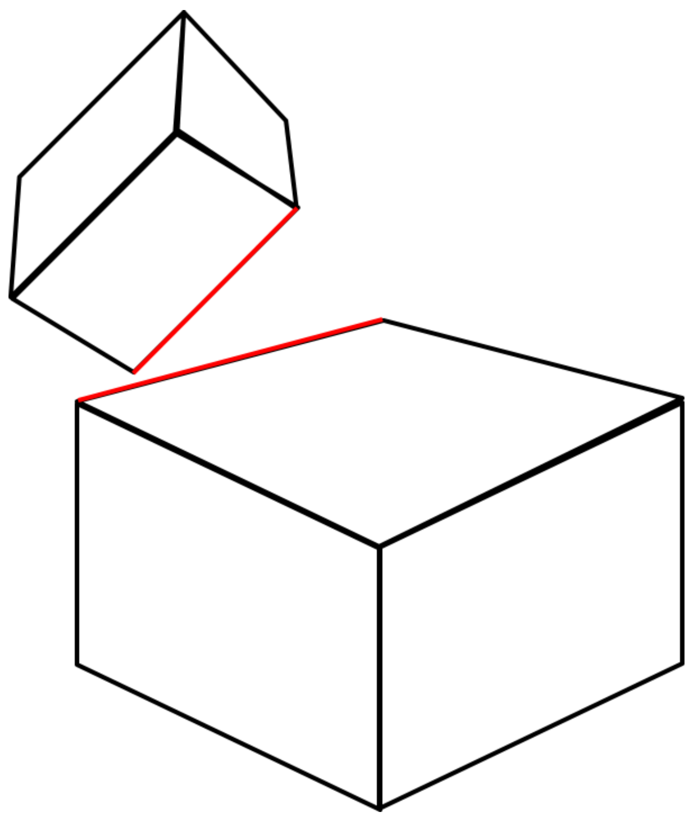
// If true, L is a separating axis perpendicular to the separating plane spanned by Az and Bz

$$\left|(T \bullet Ay)Rxz - (T \bullet Ax)Ryz\right| > |W_A Ryz| + |H_A Rxz| + |W_B Rzy| + |H_B Rzx|$$

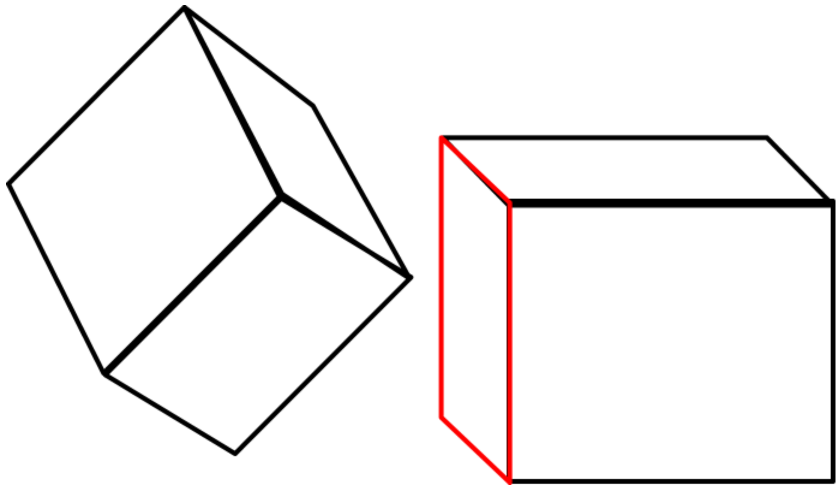
Appendix

Case Studies: Detecting Intersecting Boxes

BACK TO 15 SEPARATING AXES



Case Study 1: Consider the two boxes illustrated above. The two boxes are clearly not intersecting, yet, there is no separating plane parallel to any of the faces. However, the two boxes are not intersecting, so there must be a separating plane that exists. A separating plane that separates the two boxes is spanned by the axes of the two edges (one from each box), highlighted in red.



Case Study 2: Consider the two boxes in the illustration above. The two boxes are clearly not in collision, yet, there is no separating plane spanned by the axes of an edge from one box and an edge from the other box. However, the two boxes are not intersecting, so there must be a separating plane that separates them. A separating plane that separates the two boxes is parallel to a face (outlined in red) of one of the boxes.

BACK TO 15 SEPARATING AXES

Proof sA • (B × C) = sC • (A × B)

BACK TO CASE 7

Claim:

$$sA \bullet (B \times C) = sC \bullet (A \times B)$$

where

s = constant/scalar
A = vector
B = vector
C = vector

Proof:

$$sA \bullet (B \times C) = sA \bullet \begin{vmatrix} i & j & k \\ Bx & By & Bz \\ Cx & Cy & Cz \end{vmatrix}$$

$$= sA \bullet \left(\begin{vmatrix} By & Bz \\ Cy & Cz \end{vmatrix} i - \begin{vmatrix} Bx & Bz \\ Cx & Cz \end{vmatrix} j + \begin{vmatrix} Bx & By \\ Cx & Cy \end{vmatrix} k \right)$$

$$= s(Ax, Ay, Az) \bullet \left(\begin{vmatrix} By & Bz \\ Cy & Cz \end{vmatrix} i - \begin{vmatrix} Bx & Bz \\ Cx & Cz \end{vmatrix} j + \begin{vmatrix} Bx & By \\ Cx & Cy \end{vmatrix} k \right)$$

$$= s(Ax, Ay, Az) \bullet \left((ByCz - BzCy)i - (BxCz - BzCx)\, j + (BxCy - ByCx)k \right)$$

$$= s(Ax, Ay, Az) \bullet \left((ByCz - BzCy)i + (-BxCz + BzCx)\, j + (BxCy - ByCx)k \right)$$

$$= s(Ax, Ay, Az) \bullet \left(ByCz - BzCy, -BxCz + BzCx, BxCy - ByCx \right)$$

$$\begin{aligned} &= s(Ax(ByCz - BzCy) \\ &+ Ay(-BxCz + BzCx) \\ &+ Az(BxCy - ByCx)) \end{aligned}$$

$$\begin{aligned} &= s(AxByCz - AxBzCy \\ &- AyBxCz + AyBzCx \\ &+ AzBxCy - AzByCx) \end{aligned}$$

$$\begin{aligned} &= s(CzAxBy - CyAxBz \\ &- CzAyBx + CxAyBz \\ &+ CyAzBx - CxAzBy) \end{aligned}$$

$$\begin{aligned} &= s(CzAxBy - CzAyBx \\ &+ CyAzBx - CyAxBz \\ &+ CxAyBz - CxAzBy) \end{aligned}$$

$$\begin{aligned} &= s(CxAyBz - CxAzBy \\ &+ CyAzBx - CyAxBz \\ &+ CzAxBy - CzAyBx) \end{aligned}$$

$$\begin{aligned} &= s(Cx(AyBz - AzBy) \\ &+ Cy(AzBx - AxBz) \\ &+ Cz(AxBy - AyBx)) \end{aligned}$$

$$= sCx(AyBz - AzBy)$$
$$+ sCy(AzBx - AxBz)$$
$$+ sCz(AxBy - AyBx)$$

$$= (sCx, sCy, sCz) \bullet ((AyBz - AzBy), (AzBx - AxBz), (AxBy - AyBx))$$

$$= s(Cx, Cy, Cz) \bullet ((AyBz - AzBy), (-AxBz + AzBx), (AxBy - AyBx))$$

$$= sC \bullet ((AyBz - AzBy), -(AxBz - AzBx), (AxBy - AyBx))$$

$$= sC \bullet ((AyBz - AzBy)i - (AxBz - AzBx)j + (AxBy - AyBx)k)$$

$$= sC \bullet (\begin{vmatrix} Ay & Az \\ By & Bz \end{vmatrix} i - \begin{vmatrix} Ax & Az \\ Bx & Bz \end{vmatrix} j + \begin{vmatrix} Ax & Ay \\ Bx & By \end{vmatrix} k)$$

$$= sC \bullet \begin{vmatrix} i & j & k \\ Ax & Ay & Az \\ Bx & By & Bz \end{vmatrix}$$

$$= sC \bullet (A \times B)$$

BACK TO CASE 7

Proof T • (Ax × Bx) = (T • Az)(Ay • Bx) – (T • Ay)(Az • Bx)

BACK TO OPTIMIZING THE COMPUTATION OF T•L

Claim:

$$T \bullet (Ax \times Bx) = (T \bullet Az)(Ay \bullet Bx) - (T \bullet Ay)(Az \bullet Bx)$$

where

T = vector
Ax = unit vector representing the local x-axis of A
Ay = unit vector representing the local y-axis of A
Az = unit vector representing the local z-axis of A
Bx = unit vector representing the local x-axis of B
By = unit vector representing the local y-axis of B
Bz = unit vector representing the local z-axis of B

Note: Ax, Ay, Az are all orthogonal to one another. Similarly, Bx, By, Bz are all orthogonal to one another.

Proof:

$$T \bullet (Ax \times Bx) = T \bullet \begin{vmatrix} i & j & k \\ Axx & Axy & Axz \\ Bxx & Bxy & Bxz \end{vmatrix}$$

$$= (Tq, Tr, Ts) \bullet \left(\begin{vmatrix} Axy & Axz \\ Bxy & Bxz \end{vmatrix} i - \begin{vmatrix} Axx & Axz \\ Bxx & Bxz \end{vmatrix} j + \begin{vmatrix} Axx & Axy \\ Bxx & Bxy \end{vmatrix} k \right)$$

$$= (Tq, Tr, Ts) \bullet \left((AxyBxz - AxzBxy)i - (AxxBxz - AxzBxx)j + (AxxBxy - AxyBxx)k \right)$$

$$= (Tq, Tr, Ts) \bullet \left((AxyBxz - AxzBxy)i + (-AxxBxz + AxzBxx)j + (AxxBxy - AxyBxx)k \right)$$

$$= (Tq, Tr, Ts) \bullet \left(AxyBxz - AxzBxy, -AxxBxz + AxzBxx, AxxBxy - AxyBxx \right)$$

$$\begin{aligned} &= Tq(AxyBxz - AxzBxy) \\ &+ Tr(-AxxBxz + AxzBxx) \\ &+ Ts(AxxBxy - AxyBxx) \end{aligned}$$

$$\begin{aligned} &= Tq(-\begin{vmatrix} Ayx & Ayz \\ Azx & Azz \end{vmatrix} Bxz - \begin{vmatrix} Ayx & Ayy \\ Azx & Azy \end{vmatrix} Bxy) \\ &+ Tr(-\begin{vmatrix} Ayy & Ayz \\ Azy & Azz \end{vmatrix} Bxz + \begin{vmatrix} Ayx & Ayy \\ Azx & Azy \end{vmatrix} Bxx) \\ &+ Ts(\begin{vmatrix} Ayy & Ayz \\ Azy & Azz \end{vmatrix} Bxy - \left(- \begin{vmatrix} Ayx & Ayz \\ Azx & Azz \end{vmatrix} Bxx \right)) \end{aligned}$$

$$\begin{aligned} &= Tq(-\begin{vmatrix} Ayx & Ayz \\ Azx & Azz \end{vmatrix} Bxz - \begin{vmatrix} Ayx & Ayy \\ Azx & Azy \end{vmatrix} Bxy) \\ &+ Tr(-\begin{vmatrix} Ayy & Ayz \\ Azy & Azz \end{vmatrix} Bxz + \begin{vmatrix} Ayx & Ayy \\ Azx & Azy \end{vmatrix} Bxx) \\ &+ Ts(\begin{vmatrix} Ayy & Ayz \\ Azy & Azz \end{vmatrix} Bxy + \begin{vmatrix} Ayx & Ayz \\ Azx & Azz \end{vmatrix} Bxx) \end{aligned}$$

$$
\begin{aligned}
&= Tq(-(AyxAzz - AyzAzx)Bxz - (AyxAzy - AyyAzx)Bxy) \\
&+ Tr(-(AyyAzz - AyzAzy)Bxz + (AyxAzy - AyyAzx)Bxx) \\
&+ Ts((AyyAzz - AyzAzy)Bxy + (AyxAzz - AyzAzx)Bxx)
\end{aligned}
$$

$$
\begin{aligned}
&= Tq(-AyxAzzBxz + AyzAzxBxz - AyxAzyBxy + AyyAzxBxy) \\
&+ Tr(-AyyAzzBxz + AyzAzyBxz + AyxAzyBxx - AyyAzxBxx) \\
&+ Ts(AyyAzzBxy - AyzAzyBxy + AyxAzzBxx - AyzAzxBxx)
\end{aligned}
$$

$$
\begin{aligned}
&= Tq(-AyxAzzBxz - AyxAzyBxy + AyzAzxBxz + AyyAzxBxy) \\
&+ Tr(-AyyAzzBxz - AyyAzxBxx + AyzAzyBxz + AyxAzyBxx) \\
&+ Ts(AyyAzzBxy + AyxAzzBxx - AyzAzyBxy - AyzAzxBxx)
\end{aligned}
$$

$$
\begin{aligned}
&= Tq(-AyxAzzBxz - AyxAzyBxy) + Tq(AyzAzxBxz + AyyAzxBxy) \\
&+ Tr(-AyyAzzBxz - AyyAzxBxx) + Tr(AyzAzyBxz + AyxAzyBxx) \\
&+ Ts(AyyAzzBxy + AyxAzzBxx) + Ts(-AyzAzyBxy - AyzAzxBxx)
\end{aligned}
$$

$$
\begin{aligned}
&= TqAyx(-AzzBxz - AzyBxy) + TqAzx(AyzBxz + AyyBxy) \\
&+ TrAyy(-AzzBxz - AzxBxx) + TrAzy(AyzBxz + AyxBxx) \\
&+ TsAzz(AyyBxy + AyxBxx) + TsAyz(-AzyBxy - AzxBxx)
\end{aligned}
$$

$$
\begin{aligned}
&= -TqAyx(AzzBxz + AzyBxy) + TqAzx(AyzBxz + AyyBxy) \\
&- TrAyy(AzzBxz + AzxBxx) + TrAzy(AyzBxz + AyxBxx) \\
&+ TsAzz(AyyBxy + AyxBxx) - TsAyz(AzyBxy + AzxBxx)
\end{aligned}
$$

$$
\begin{aligned}
&= -TqAyx(AzzBxz + AzyBxy) + TqAzx(AyzBxz + AyyBxy) \\
&- TrAyy(AzzBxz + AzxBxx) + TrAzy(AyzBxz + AyxBxx) \\
&- TsAyz(AzyBxy + AzxBxx) + TsAzz(AyyBxy + AyxBxx)
\end{aligned}
$$

$$
\begin{aligned}
&= -TqAyx(AzyBxy + AzzBxz) + TqAzx(AyyBxy + AyzBxz) \\
&- TrAyy(AzxBxx + AzzBxz) + TrAzy(AyxBxx + AyzBxz) \\
&- TsAyz(AzxBxx + AzyBxy) + TsAzz(AyxBxx + AyyBxy)
\end{aligned}
$$

$$
\begin{aligned}
&= -TqAyx(AzyBxy + AzzBxz) + TqAzx(AyyBxy + AyzBxz) + 0 \\
&- TrAyy(AzxBxx + AzzBxz) + TrAzy(AyxBxx + AyzBxz) + 0 \\
&- TsAyz(AzxBxx + AzyBxy) + TsAzz(AyxBxx + AyyBxy) + 0
\end{aligned}
$$

$$= -TqAyx(AzyBxy + AzzBxz) + TqAzx(AyyBxy + AyzBxz) - TqAyxAzxBxx + TqAyxAzxBxx$$
$$- TrAyy(AzxBxx + AzzBxz) + TrAzy(AyxBxx + AyzBxz) - TrAyyAzyBxy + TrAyyAzyBxy$$
$$- TsAyz(AzxBxx + AzyBxy) + TsAzz(AyxBxx + AyyBxy) - TsAyzAzzBxz + TsAyzAzzBxz$$

$$= -TqAyx(AzyBxy + AzzBxz) - TqAyxAzxBxx + TqAzx(AyyBxy + AyzBxz) + TqAyxAzxBxx$$
$$- TrAyy(AzxBxx + AzzBxz) - TrAyyAzyBxy + TrAzy(AyxBxx + AyzBxz) + TrAyyAzyBxy$$
$$- TsAyz(AzxBxx + AzyBxy) - TsAyzAzzBxz + TsAzz(AyxBxx + AyyBxy) + TsAyzAzzBxz$$

$$= -TqAyx(AzxBxx + AzyBxy + AzzBxz) + TqAzx(AyxBxx + AyyBxy + AyzBxz)$$
$$- TrAyy(AzxBxx + AzyBxy + AzzBxz) + TrAzy(AyxBxx + AyyBxy + AyzBxz)$$
$$- TsAyz(AzxBxx + AzyBxy + AzzBxz) + TsAzz(AyxBxx + AyyBxy + AyzBxz)$$

$$= -TqAyx(AzxBxx + AzyBxy + AzzBxz)$$
$$+ TqAzx(AyxBxx + AyyBxy + AyzBxz)$$
$$- TrAyy(AzxBxx + AzyBxy + AzzBxz)$$
$$+ TrAzy(AyxBxx + AyyBxy + AyzBxz)$$
$$- TsAyz(AzxBxx + AzyBxy + AzzBxz)$$
$$+ TsAzz(AyxBxx + AyyBxy + AyzBxz)$$

$$= -TqAyx(AzxBxx + AzyBxy + AzzBxz)$$
$$- TrAyy(AzxBxx + AzyBxy + AzzBxz)$$
$$- TsAyz(AzxBxx + AzyBxy + AzzBxz)$$
$$+ TqAzx(AyxBxx + AyyBxy + AyzBxz)$$
$$+ TrAzy(AyxBxx + AyyBxy + AyzBxz)$$
$$+ TsAzz(AyxBxx + AyyBxy + AyzBxz)$$

$$= (-TqAyx - TrAyy - TsAyz)(AzxBxx + AzyBxy + AzzBxz)$$
$$+ (TqAzx + TrAzy + TsAzz)(AyxBxx + AyyBxy + AyzBxz)$$

$$= (-TqAyx - TrAyy - TsAyz)((Azx, Azy, Azz) \bullet (Bxx, Bxy, Bxz))$$
$$+ (TqAzx + TrAzy + TsAzz)((Ayx, Ayy, Ayz) \bullet (Bxx, Bxy, Bxz))$$

$$= (-TqAyx - TrAyy - TsAyz)(Az \bullet Bx)$$
$$+ (TqAzx + TrAzy + TsAzz)(Ay \bullet Bx)$$

$$= -(TqAyx + TrAyy + TsAyz)(Az \bullet Bx)$$
$$+ (TqAzx + TrAzy + TsAzz)(Ay \bullet Bx)$$

$$= (TqAzx + TrAzy + TsAzz)(Ay \bullet Bx)$$
$$- (TqAyx + TrAyy + TsAyz)(Az \bullet Bx)$$

$$= ((Tq, Tr, Ts) \bullet (Azx, Azy, Azz))(Ay \bullet Bx)$$
$$- ((Tq, Tr, Ts) \bullet (Ayx, Ayy, Ayz))(Az \bullet Bx)$$

$$= (T \bullet Az)(Ay \bullet Bx)$$
$$- (T \bullet Ay)(Az \bullet Bx)$$

$$= (T \bullet Az)(Ay \bullet Bx) - (T \bullet Ay)(Az \bullet Bx)$$

BACK TO OPTIMIZING THE COMPUTATION OF T•L

Bibliography

Gomez, Miguel. "Simple Intersection Tests For Games." 18 October 1999. Gamasutra. 27 November 2008 <http://www.gamasutra.com/features/19991018/Gomez_1.htm>.

S. Gottschalk, M. Lin, and D. Manocha. “OBB-Tree: A Hierarchical Structure for Rapid Interference Detection.” ACM SIGGRAPH. 1996.
(Downloaded from http://www.stanford.edu/class/cs273/refs/obb.pdf)