Intersection of a Triangle and a Cylinder

David Eberly, Geometric Tools, Redmond WA 98052
https://www.geometrictools.com/

Created: November 2, 2010
Last Modified: April 30, 2021

Contents

1 Introduction

The (solid) triangle has vertices $\mathbf{P}_i$ for $0 \le i \le 2$ and is parameterized using barycentric coordinates by $b_0\mathbf{P}_0 + b_1\mathbf{P}_1 + b_2\mathbf{P}_2$ for $b_0 \in [0,1]$, $b_1 \in [0,1]$, $b_2 \in [0,1]$ and $b_0 + b_1 + b_2 = 1$. The edges are parameterized with a single parameter. For example, the edge connecting $\mathbf{P}_0$ and $\mathbf{P}_1$ is parameterized by $(1-t)\mathbf{P}_0 + t\mathbf{P}_1$ for $t \in [0,1]$.

The (solid) cylinder has center $\mathbf{C}$, unit-length axis direction $\mathbf{D}$, radius r, and height h. Choose unit-length vectors $\mathbf{U}$ and $\mathbf{V}$ so that $\{\mathbf{U}, \mathbf{V}, \mathbf{D}\}$ is an orthonormal set; that is, the vectors are unit length and mutually perpendicular. The implementation of the algorithm does not depend on whether the set is right handed or left handed. A parameterization is $\mathbf{C} + x\mathbf{U} + y\mathbf{V} + z\mathbf{D}$, where $x^2 + y^2 \le r^2$ and $|z| \le h/2$.

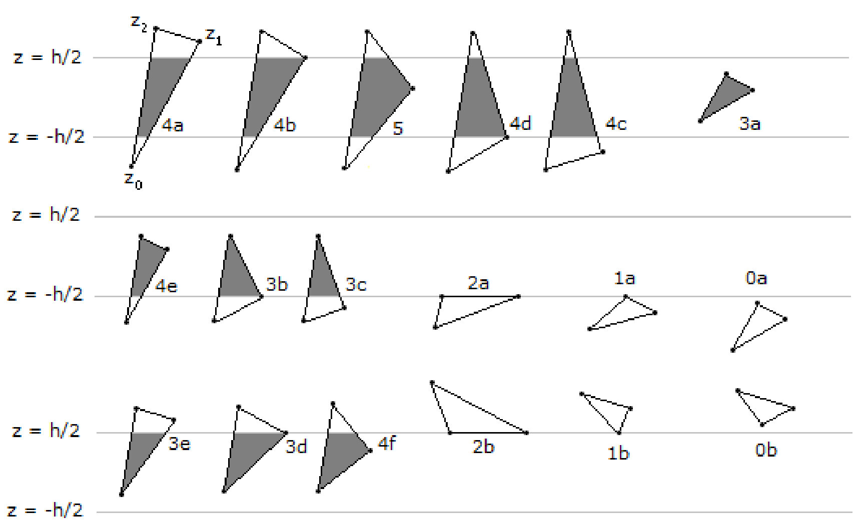
Both objects are assumed to be stationary, and the query is a *test-intersection query* whose result is a Boolean indicating whether or not the objects intersect.

2 The Algorithm

The problem can be modified by transforming space so that the cylinder has center $(0,0,0)$ and direction $(0,0,1)$. The parameterization is (x,y,z) where $x^2 + y^2 \le r^2$ and $|z| \le h/2$. Compute the representations of $\mathbf{P}_i$ in transformed coordinates; that is, $\mathbf{P}_i = \mathbf{C} + x_i\mathbf{U} + y_i\mathbf{V} + z_i\mathbf{D}$, where $x_i = \mathbf{U} \cdot (\mathbf{P}_i - \mathbf{C})$, $y_i = \mathbf{V} \cdot (\mathbf{P}_i - \mathbf{C})$ and $z_i = \mathbf{D} \cdot (\mathbf{P}_i - \mathbf{C})$. Define $\mathbf{Q}_i = (x_i, y_i, z_i)$ for all i. We want to determine whether the triangle with vertices $\mathbf{Q}_i$ intersects the cylinder.

Clip the triangle to the slab $|z| \le h/2$; that is, clip against two planes, $z = h/2$ and $z = -h/2$. The result is one of: the empty set, a single point, a line segment, a triangle, a convex quadrilateral or a convex pentagon. Before clipping, sort the $\mathbf{Q}_i$ so that $z_0 \le z_1 \le z_2$. Figure 1 shows the possibilities for the relationship between the triangle and the slab.

Figure 1. The relationship between the triangle and the z-slab.



The pseudocode for distinguishing among the cases is shown in Listing 1.

Listing 1. Pseudocode for clipping against the slab $-h/2 \le z \le h/2$ is shown here. Each case has an associated block of code to test whether the xy-projection of the clipped triangle overlaps the disk of projection of the cylinder.

```
if (z2 < -h/2) { case 0a }
if (z0 >  h/2) { case 0b }
if (-h/2 <= z0 && z2 <= h/2) { case 3a }

if (z0 < -h/2)
{
    if (z2 > h/2)
    {
        if (z1 >= h/2) { cases 4a/4b }
        else if (z1 <= -h/2) { cases 4c/4d }
        else { case 5 }
    }
    else if (z2 > -h/2)
    {
        if (z1 <= -h/2) { cases 3b/3c }
        else { case 4e }
    }
    else
    {
        if (z1 < -h/2) { case 1a }
        else { case 2a }
    }
}
else if (z0 < h/2)
{
    if (z1 >= h/2) { cases 3d/3e }
    else { case 4f }
}
else
{
    if (z1 > h/2) { case 1b }
    else { case 2b }
}
```

To illustrate the clipping, consider Case 4a of Figure 1. The edge $\langle \mathbf{P}_0, \mathbf{P}_2 \rangle$ is parameterized by $\mathbf{P}_0 + t(\mathbf{P}_2 - \mathbf{P}_0)$ for $t \in [0, 1]$. On the bottom of the slab, $-h/2 = z_0 + t(z_1 - z_0)$ and has solution $t_0 = (-h/2 - z_0)/(z_2 - z_0)$. On the top of the slab, $+h/2 = z_0 + t(z_2 - z_0)$ and has solution $t_3 = (+h/2 - z_0)/(z_2 - z_0)$. The edge $\langle \mathbf{P}_0, \mathbf{P}_1 \rangle$ is parameterized by $\mathbf{P}_0 + t(\mathbf{P}_1 - \mathbf{P}_0)$ for $t \in [0, 1]$. On the bottom of the slab, $-h/2 = z_0 + t(z_1 - z_0)$ and has solution $t_1 = (-h/2 - z_0)/(z_1 - z_0)$. On the top of the slab, $+h/2 = z_0 + t * (z_1 - z_0)$ and has solution $t_2 = (+h/2 - z_0)/(z_1 - z_0)$. The clipped triangle is a convex quadrilateral with ordered vertices $\mathbf{Q}_0 = \mathbf{P}_0 + t_0(\mathbf{P}_2 - \mathbf{P}_0)$, $\mathbf{Q}_1 = \mathbf{P}_0 + t_1(\mathbf{P}_1 - \mathbf{P}_0)$, $\mathbf{Q}_2 = \mathbf{P}_0 + t_2(\mathbf{P}_1 - \mathbf{P}_0)$ and $\mathbf{Q}_3 = \mathbf{P}_0 + t_3(\mathbf{P}_2 - \mathbf{P}_0)$. The projection of the triangles uses the xy-coordinates of these points.

Generally, if the clipped and projected triangle is a point (x_0, y_0), then intersection occurs when $x_0^2 + y_0^2 <= r^2$.

If the clipped and projected triangle is a line segment with endpoints (x_0, y_0) and (x_1, y_1), compute the squared distance, d^2 from $(0, 0)$ to the segment. Intersection occurs when $d^2 <= r^2$. The squared distance is computed only when needed. If $x_0^2 + y_0^2 \le r^2$, then (x_0, y_0) is in the disk, so the triangle and cylinder intersect. If $x_1^2 + y_1^2 \le r^2$, then (x_1, y_1) is in the disk, so the triangle and cylinder intersect. If neither endpoint is in the disk, it is possible the segment and disk intersect. At this time the squared distance is computed:

$$d^2 = \frac{|(x_0 - x_1, y_0 - y_1) \cdot (x_0, y_0)|^2}{|(x_1, y_1)|^2} \tag{1}$$

Otherwise, the clipped and projected triangle is a convex polygon (triangle, quadrilateral or pentagon). If $(0, 0)$ is contained in the polygon, there is intersection. The point-in-convex-polygon test is straightforward, testing the relative location of $(0, 0)$ to each edge of the polygon. If $(0, 0)$ is not contained in the polygon, compute the squared distance d^2 from $(0, 0)$ to each edge of the polygon. If any of the d^2 is less than or equal to r^2, there is intersection.

3 The Implementation

The test-intersection query is in

GeometricTools/GTE/Mathematics/IntrTriangle3Cylinder3.h

A sample application to test the code is in the folder

GeometricTools/GTE/Samples/Intersection/IntersectTriangleCylinder

A red cylinder and a blue triangle are displayed. You can select the object to move by pressing keys. Press '0' to move the triangle or '1' to move the cylinder. The x/X, y/Y and z/Z keys translate the object. The p/P, r/R and h/H keys rotate the object. Whenever the triangle and cylinder intersect, the triangle changes color to green.

The test-intersection query can also be formulated as the minimization of squared distance between pairs of points, each pair consisting of a triangle point and a cylinder point. Generally, this is a convex quadratic programming problem with linear inequality constraints. Specifically, minimize $x^2 + y^2$ subject to 1 equality constraint (triangle is in a plane), 3 inequality constraints (point is in triangle), and 2 inequality constraints (point is between bounding planes of top/bottom of cylinder). If the minimum d^2 is less than or equal to r^2, then the triangle and cylinder intersect. A Linear Complementarity Problem (LCP) solver can be used for the minimization. See Section 7.4.3 of Robust and Error-Free Geometric Computing, 1st edition for details.