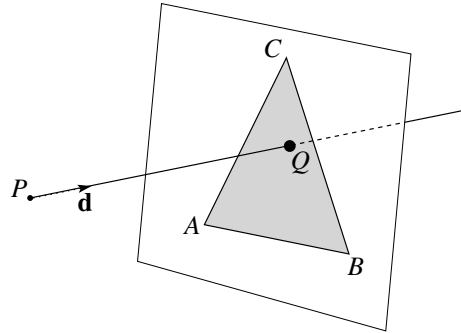


Ray-triangle intersection

Brian Curless

In this handout, we explore the steps needed to compute the intersection of a ray with a triangle, and then to compute the barycentric coordinates of that intersection. First, we consider the geometry of such an intersection:



where a ray with origin P and direction \mathbf{d} intersects a triangle defined by its vertices, A , B , and C at intersection point Q . The square region diagrammatically surrounding triangle ABC represents the *supporting plane* of the triangle, i.e., the plane that the triangle lies on.

To perform ray-triangle intersection, we must perform two steps:

1. Determine the point of intersection, Q .
2. Determine if Q lies inside of triangle ABC .

Finally, we'll compute the barycentric coordinates of Q with respect to ABC .

Determining Q

Determining the point of intersection Q in turn requires us to take two more steps:

1. Determine the equation of the supporting plane for triangle ABC .
2. Intersect the ray with the supporting plane.

We'll handle these steps in reverse order.

Ray-plane intersection

It is well known that the equation of a plane can be written as:

$$ax + by + cz = d$$

The coefficients a , b , and c form a vector that is normal to the plane, $\mathbf{n} = [a \ b \ c]^T$. Thus, we can re-write the plane equation as:

$$\mathbf{n} \cdot \mathbf{x} = d$$

where $\mathbf{x} = [x \ y \ z]^T$.

Now we consider the ray determined by P and \mathbf{d} :

$$R(t) = P + t\mathbf{d}$$

To solve for the intersection of ray $R(t)$ with the plane, we simply substitute $\mathbf{x} = R(t)$ into the plane equation and solve for t :

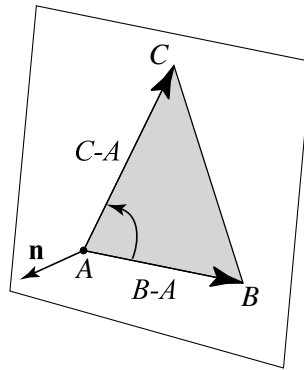
$$\begin{aligned} \mathbf{n} \cdot R(t) &= d \\ \mathbf{n} \cdot [P + t\mathbf{d}] &= d \\ \mathbf{n} \cdot P + t\mathbf{n} \cdot \mathbf{d} &= d \\ t &= \frac{d - \mathbf{n} \cdot P}{\mathbf{n} \cdot \mathbf{d}} \end{aligned}$$

Note that if $\mathbf{n} \cdot \mathbf{d} = 0$, then \mathbf{d} is parallel to the plane and the ray does not intersect the plane (i.e., the intersection is at infinity). Be sure to check for this case!

The intersection point Q can thus be determined by plugging the resulting t into the ray equation for $R(t)$.

Solving for the supporting plane

To intersect with a triangle, we need to solve for the equation of its supporting plane. As noted above, we need only solve for the normal \mathbf{n} to the plane and the coefficient d . The figure below illustrates how we might solve for the normal using cross products.



In particular, we can construct the vectors $B-A$ and $C-A$ and compute their normalized cross product:

$$\mathbf{n} = \frac{(B - A) \times (C - A)}{\|(B - A) \times (C - A)\|}$$

[Strictly speaking, \mathbf{n} does not need to be normalized (i.e., of unit length) in order for ray-plane and ray-triangle intersection to work; however, for shading purposes, a normalized normal is necessary anyway, so we present the normalized result.]

We should take a moment here to make a few comments. First, if we view the normal to a triangle as simply being a unit vector that is perpendicular to the supporting plane, then, in fact, both the \mathbf{n} computed above and $-\mathbf{n}$ would satisfy our definition. Which one is “right”?

We'll assume that a triangle is actually "one-sided" so that there is a correct normal direction that points to the outside of the object that the triangle belongs to. In addition, we'll assume that this direction is implicit in the ordering of the vertices. In particular, if you curl the fingers of your right hand from A to B to C , then your thumb will point in the direction of the normal. This right-hand rule is another way of saying that we expect the vertices to be arranged in counter-clockwise order with respect to the normal.

Next, we chose to cross $B-A$ into $C-A$. Did the order matter? Yes! If we had done it the other way around, we would have been computing $-\mathbf{n}$.

Finally, could we have chosen other vectors for computing the normal? Definitely! Any two vectors that are tangent to the plane can be crossed into each other (in the correct order!) to yield the normal vector. For instance, we could have crossed $C-B$ with $A-B$, in that order, to compute the normal.

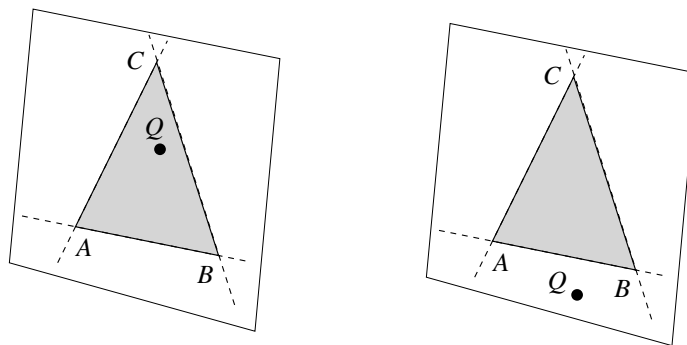
Getting back to the sub-problem at hand, how do we compute the remaining coefficient d in order to establish the supporting plane equation? Earlier we stated the plane equation as $\mathbf{n} \cdot \mathbf{x} = d$. Any \mathbf{x} that satisfies this equation must lie on the plane. Thus, if we pick a point that lies on the plane, and plug it into the equation, then the equation must be satisfied. Clearly, the vertices of the triangle lie on the supporting plane, so we just need to pick one – *any* one – of them. We'll pick A . Now we find that, simply:

$$d = \mathbf{n} \cdot A$$

At this point, we have now computed all the coefficients needed to describe the supporting plane of the triangle and can go ahead and compute the intersection Q .

Triangle inside-outside testing

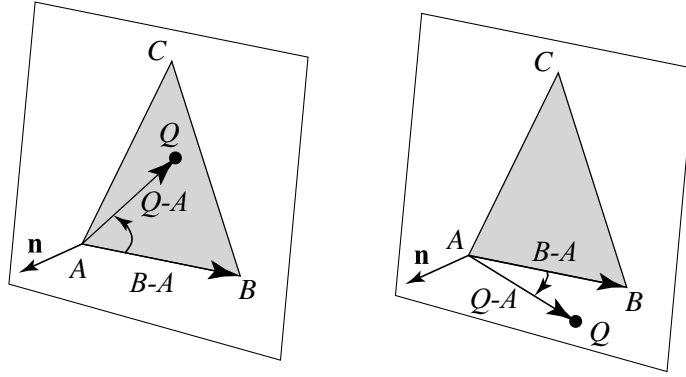
Now we've got the situation where we have a triangle ABC and an intersection point Q , but we don't know if Q actually lies inside the triangle. To determine this, we need to perform an "inside-outside" test. In particular, Q lies inside of triangle ABC if it lies on the inside of all of the lines determined by edges AB , BC , and CA . Below, on the left, we see that Q satisfies this requirement. On the right, we see that Q is inside of edges CA and BC , but not AB , and is therefore outside the triangle.



We can now use cross products to determine which side of each edge Q is on. Consider the edge AB . If we form the vectors $B-A$ and $Q-A$, then their cross product should point in the same direction as the normal. In other words, Q is on the inside of AB if:

$$[(B - A) \times (Q - A)] \cdot \mathbf{n} \geq 0$$

The figure below, on the left, illustrates this idea for the two cases where Q is inside and outside of the edge AB . In the second case, the cross product will point in the opposite direction of the normal.



We can develop similar conditions for the other edges. As a result, Q is inside of ABC if all of the following conditions are met simultaneously:

$$\begin{aligned} [(B - A) \times (Q - A)] \cdot \mathbf{n} &\geq 0 \\ [(C - B) \times (Q - B)] \cdot \mathbf{n} &\geq 0 \\ [(A - C) \times (Q - C)] \cdot \mathbf{n} &\geq 0 \end{aligned}$$

If any one of these tests fails, then Q is not inside of ABC , and the result is no intersection.

Note that we used the “greater than or equal to” comparison here. We do this in order to insure that the vertices A , B , and C and the edges AB , BC , and CA are all considered to be part of the triangle. For instance if Q were equal to A , then the first test would return a value of zero. Or, if Q were along the edge BC , then the second test would return a value of zero (parallel vectors have a cross product of zero). In both cases, however, we would want to declare Q to have intersected with the triangle; thus, the tests must have the “greater than or equal to” comparisons.

Note that the triangle-intersection procedure described in this document is not as efficient as it could be. As noted in class, you can speed up the procedure by first projecting the triangle onto an axis aligned plane and then perform the calculations in 2D. The details of this procedure are beyond the scope of this document.

Computing barycentric coordinates

As a final step, if Q is inside of ABC , we will compute the barycentric coordinates of Q with respect to ABC . Recall from lecture that the barycentric coordinates can be calculated in terms of area ratios:

$$\alpha = \frac{\text{Area}(QBC)}{\text{Area}(ABC)} \quad \beta = \frac{\text{Area}(AQC)}{\text{Area}(ABC)} \quad \gamma = \frac{\text{Area}(ABQ)}{\text{Area}(ABC)}$$

In fact, these areas are each proportional to the lengths of the corresponding cross products described above. For instance,

$$\text{Area}(QBC) = \|(C - B) \times (Q - B)\| / 2$$

Since the cross product points in the \mathbf{n} direction (given that Q is inside of ABC), then we can actually compute the length of the cross product as simply:

$$\|(C - B) \times (Q - B)\| = [(C - B) \times (Q - B)] \cdot \mathbf{n}$$

Finally, we have the equations for the barycentric coordinates of Q :

$$\alpha = \frac{[(C-B) \times (Q-B)] \cdot \mathbf{n}}{[(B-A) \times (C-A)] \cdot \mathbf{n}} \quad \beta = \frac{[(A-C) \times (Q-C)] \cdot \mathbf{n}}{[(B-A) \times (C-A)] \cdot \mathbf{n}} \quad \gamma = \frac{[(B-A) \times (Q-A)] \cdot \mathbf{n}}{[(B-A) \times (C-A)] \cdot \mathbf{n}}$$

Note that we already computed the numerators in the previous sections, as well as the terms needed for the denominator, so very little extra work is needed to establish the barycentric coordinates.

Once these coordinates are calculated it is possible to, e.g., interpolate normals stored at vertices. In that case, one would compute the normal \mathbf{N}_Q at point Q to be a weighted sum of the normals \mathbf{N}_A , \mathbf{N}_B , and \mathbf{N}_C at vertices A , B , and C , respectively:

$$\mathbf{N}_Q = \frac{\alpha \mathbf{N}_A + \beta \mathbf{N}_B + \gamma \mathbf{N}_C}{\|\alpha \mathbf{N}_A + \beta \mathbf{N}_B + \gamma \mathbf{N}_C\|}$$

The denominator is required so that \mathbf{N}_Q is of unit length and can be used in subsequent shading, reflection, and refraction calculations.

Speeding it up

The approach described so far is perfectly good for doing triangle intersection and computing barycentric coordinates. However, it can be made to run faster. (This is slightly trickier and is optional!)

Note that barycentric coordinates are preserved by affine transformations. So, after computing Q , we can orthographically project out a dimension – e.g., drop the z -coordinate – to convert A , B , C , and Q to A' , B' , C' , and Q' , where the primed versions are now 2D points (e.g., with only x and y coordinates after projecting out the z -coordinate). Now you can do the inside-outside test and barycentric coordinate calculations in 2D; the 2D test result and barycentric coordinates will be identical to what you would get in the 3D case. But, you get a significant computational savings. E.g., computing cross products in 2D will be significantly cheaper.

Which axis would you project away? Suppose your triangle were parallel to the x - z plane. In that case, if you were to project out the z -axis, the triangle would project down to a line segment, i.e., a degenerate triangle with zero area. You would not be able to reliably perform an inside-out test or compute barycentric coordinates in this case. If instead your triangle were just *nearly* parallel to the x - z plane, then the z -projected triangle would be *nearly* degenerate, which would still be undesirable numerically. For this triangle, the best bet would be to project out the y -axis, to give a projected triangle that is as big as possible (as non-degenerate as possible). In general, the best axis to project away can be determined from the normal: project away the axis corresponding to the component of the normal that is largest.