Introduction to Computer Graphics
Dr. Prem Kalra
Department of Computer Science and Engineering
Indian Institute of Technology, Delhi
Lecture - 24
Ray Tracing

We have been talking about ray tracing. The idea here is that we want to simulate a process which we observe in reality, how the object of the scene is illuminated. So typically what you have seen is that if you have a light source here and there are objects in the scene and this is your camera so the way we see is that the interaction of the light source happens through the rays which are coming from the light source and whatever gets intercepted through the camera here is what we observe. That is the process of rendering.

Now the problem which we observe here is that there are too many rays which coming out of a light source or many light sources which may or may not contribute to the formation of the image just because of the fact that many rays may not get intercepted here. So the effort which is being applied here is perhaps too much for what we just count for the image. So this is not perhaps an efficient way of generating an image. So what we do is we reverse the process.

This is called as forward ray tracing. Then if I reverse the process, it is backward ray tracing where we start from the image plane itself, the pixels of the image plane themselves. So there is ray which is emanated or shot from the eye of the viewer for each pixel in the image plane and then depending on the nature of the objects or the material of the surface of the object there will be a certain behavior to that ray which intersects the object so it may get reflected it may get refracted and so on.

(Refer Slide Time: 00:03:56)

The idea here is that we would trace that ray to decide the illumination of the point which is intersected by the ray. This backward ray tracing is actually nothing but ray tracing.

Actually there are two issues when we are talking about rendering. One of them is that we should be able to decide what point is visible. So it is determining the visibility of the scene is one issue and that we can ascertain by considering the closest point with respect to the viewer given as a consequence of looking at the intersections with the scene of the ray. Then you consider the nearest point which is intersected. That is how we established visibility in the scene. That is one issue which we resolve through ray tracing. The other issue is to determine the illumination at the point of intersection which we have ascertained to be visible. There we just applied the illumination model. Depending on the material of the object the ray which comes to the object may get reflected or may get refracted.

(Refer Slide Time: 00:07:02)

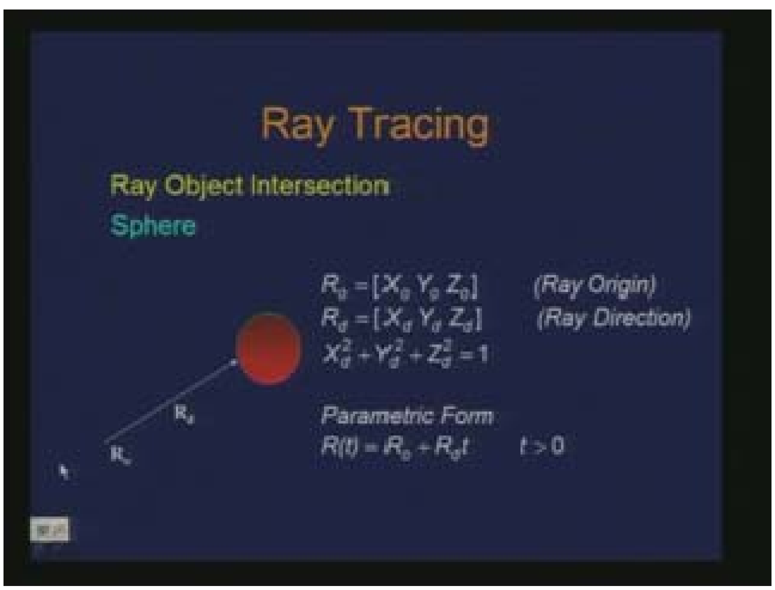
If I discard those so called secondary rays, these are the secondary rays, then we are talking about only one level of ray tracing of the rays, which is the ray which gets off the viewer to the object or the scene and wherever it intersects the first point of intersection with respect to the viewer, that is the point to be rendered. That is a simplified scenario where we discard the further secondary rays beyond the point of intersection. That is called as ray casting or one level ray tracing. So when we do this the issue is on finding the ray of intersection, this is one task and the other thing is apply the illumination model.

Ray object intersection:
Last time we looked at ray sphere intersection. These are the two issues; the visibility test and the illumination model. These are the two issues related to ray tracing. When we look at ray object intersection with the sphere as an object we consider a parametric form of the ray. Ray is defined as the origin of ray which is a point R_0 and the direction of ray which is defined as R_D and the parameter t. This is the line definition.

(Refer Slide Time: 00:07:17)

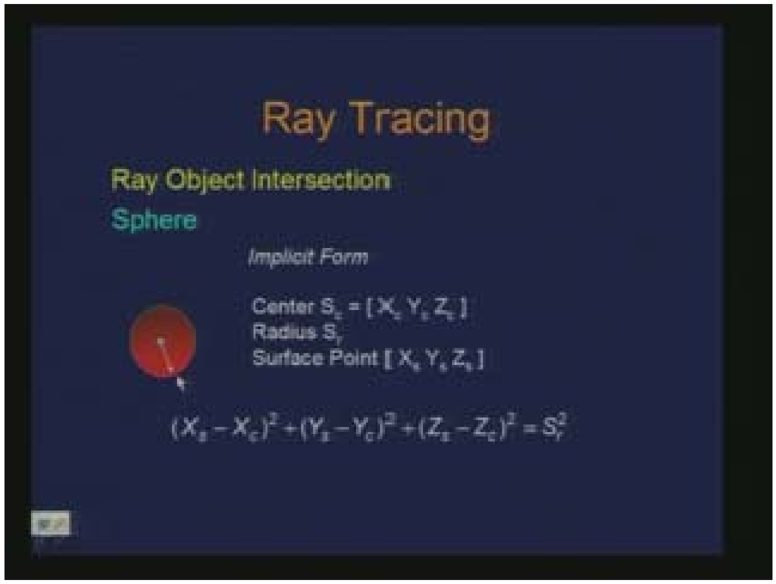
Therefore, clearly we use t greater than 0 because we are considering that we will look at the rays which will emanate from the origin of the ray which is the viewer or the eye towards the scene. We are not concerned about the rays which are in the opposite direction. Therefore only positive things have to be considered.

(Refer Slide Time: 00:08:31)



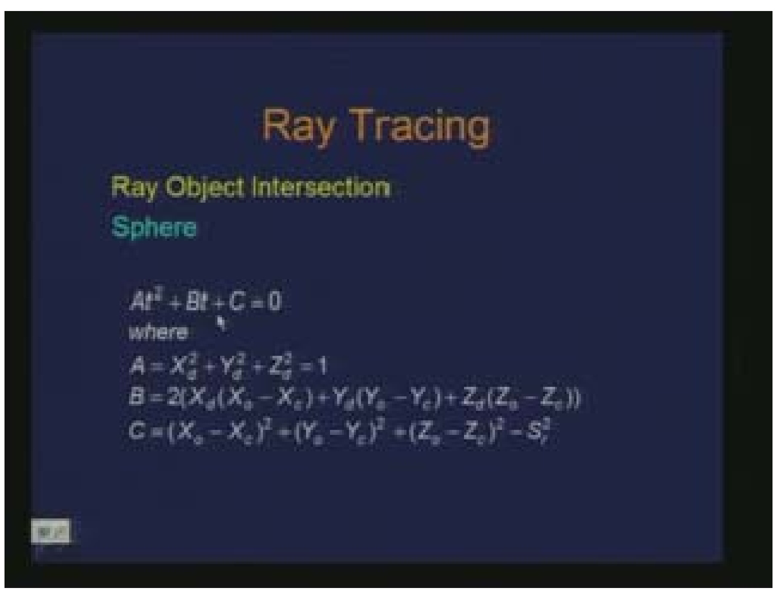
For the purpose of the simplicity we may consider that the direction of the ray to be normalized so we have the unit vector. This is how we look at the definition of the ray which is in parametric form. As far as the definition of the sphere is concerned we consider this as an implicit form where we define the sphere through the center and the radius of the sphere.

(Refer Slide Time: 00:09:32)



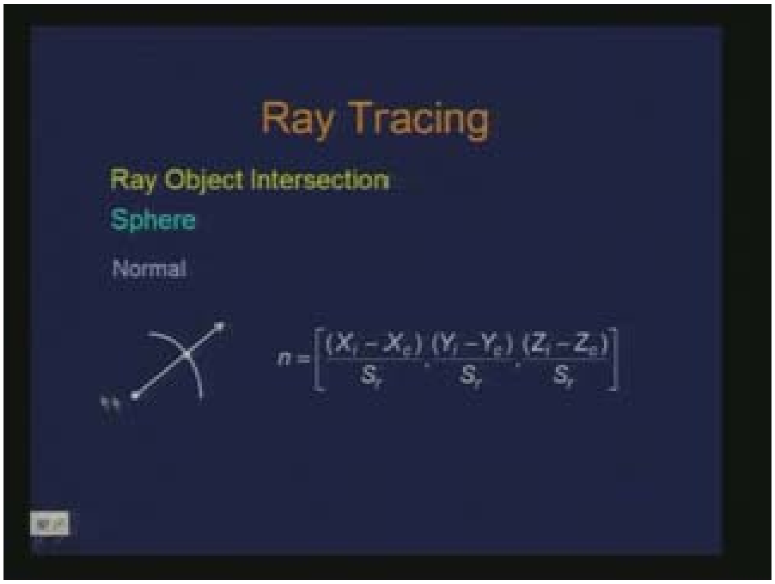
We have an implicit equation for the sphere which is (Xs minus Xc) power 2 plus (Ys minus Yc) power 2 plus (Zs minus Zc) power 2 is equal to Sr power 2. So Xs, Ys, Zs is any surface point on the sphere.

(Refer Slide Time: 00:09:58)



Now obtaining the intersection of the ray with sphere turns out to be substituting the equation of the ray into this equation and solve for t. That is what it becomes. This actually gives you a quadratic equation in t in the form of At power 2 plus Bt plus C is equal to 0 if you substitute the rays equation into the spheres equation where these are the coefficients A B C in the known quantity in the quantities you know. So the solution of this could be found. There will be two values to t. So we will consider the smallest value of t because we are looking at the closest point. Now, given the point of intersection here X_i Y_i Z_i the other problem comes is to find out the normal at that point.

(Refer Slide Time: 00:11:32)



So in the case of sphere it is a very simple situation. Just the radial direction from the point gives you the normal. So this is the normal which is X_i minus X_c so it is X_c Y_c Z_c are the coordinates of the center and S_r is the radius. This is the unit normal vector at the point of intersection which you have computed. Calculation of normal is essential because we would like to compute the illumination at that point which comes from the normal. So this was a very simple way of obtaining intersection of ray to sphere. There is also an alternate way to find out the intersection with the sphere which is somewhat a geometric approach.

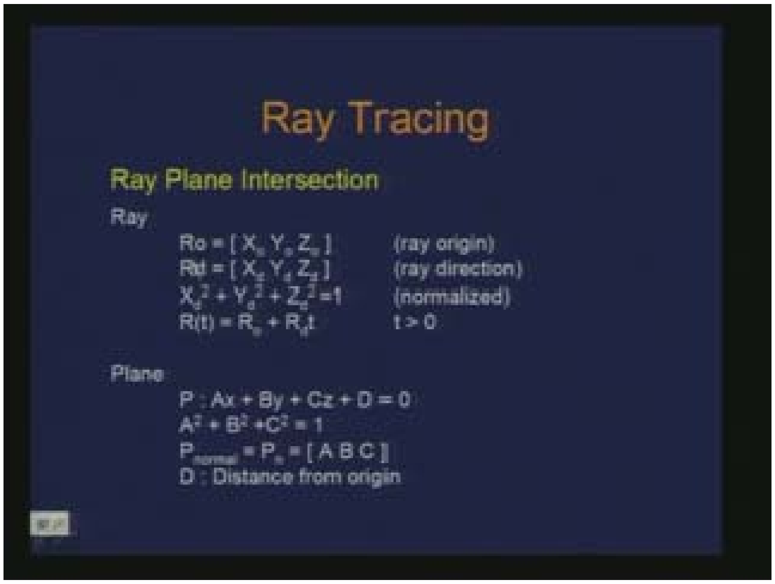
Consider this as a sphere with a center here O, this is your origin of the ray R_0, this is some vector L and this is the direction of the ray R_D which goes from here so in effect I am trying to find out these green points and I will select this being the closest. If I form this line there through R_0 in the direction of Rd this particular length is nothing but the radius of the sphere.

(Refer Slide Time: 00:17:01)

This distance is the distance d from the origin of the sphere to the redirection and this particular distance which is traversed along the ray direction is tca and this distance is thc. So if I can somehow relate this distance in terms of tca and thc I should be able to get the ts which should give me the point of intersection. Now the question is whether I can compute these tca and thc. So what we are saving is this l is nothing but O minus R_0. What is tca? It is a projection of l to this direction of the ray which is nothing but a dot product which is also l transpose times R_D so I can compute tca and I can compute l. Now there are also cases where I can do some elimination of further computation.

If I know this tca is less than 0 that is nothing but saying that ts are positive then there is no interaction, it is on the other side of the ray so I can compute tca. Now how about d? This d is nothing but this l power 2 minus tca square. I have already computed tca, l is ltl so I can compute d. Again if t is greater than r there is no interception. The next thing is to get the thc. So thc is nothing but square root r power 2 minus d power 2 from this triangle then t of interest is tca minus thc and tca plus thc these are the two ts t_0 and t_1 and then you take the smaller t. So you can also do the same computation which we did algebraically by substituting the rays equation into the spheres equation and solving for t. You can also do that in geometric form using geometry.

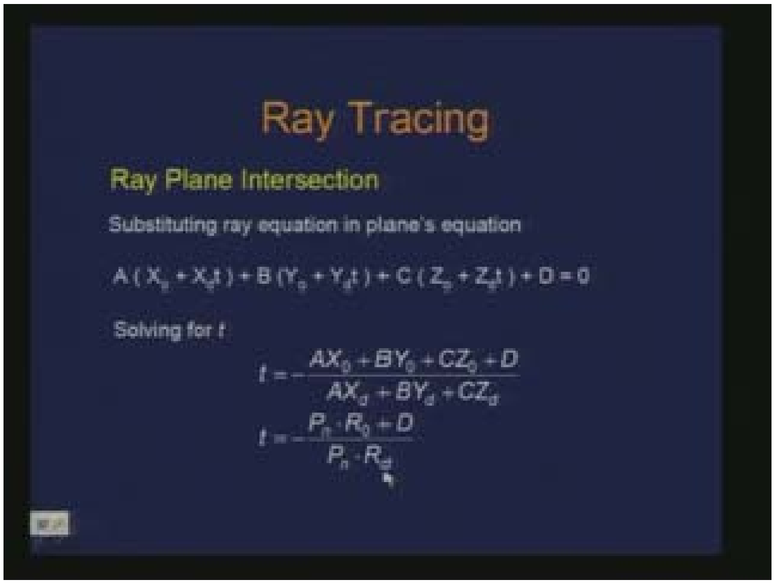
(Refer Slide Time: 00:18:19)



So this is the sphere ray intersection. In a similar fashion we can now look at other kinds of objects. Here we have ray plane intersection. So once again I consider ray to be defined in a parametric form exactly the same way as I considered for sphere and this plane is again defined in an implicit form A_x plus B_y plus C_z plus D is equal to 0 and I can assert that A B C gives me the unit vector so A power 2 plus B power 2 plus C power 2 is equal to 1 which in turn also defines the normal of the plane so I do not have to explicitly compute the normal of the plane because I know from the equation of the plane itself and d is nothing but the distance from the origin. So this is the equation of the plane I have.

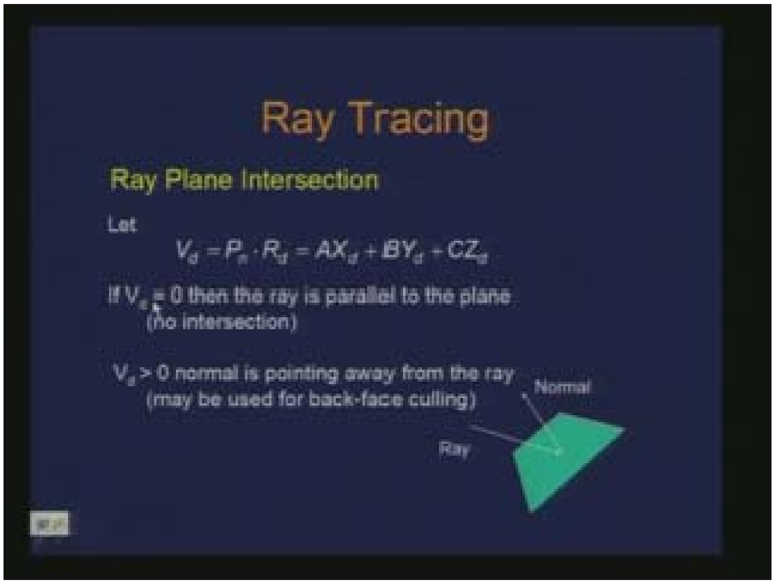
As we exactly did in the case of sphere I can substitute this ray equation into this and solve for t. This is what is done here, you substitute ray equation in planes equations and solve for t. So when you look at the solution we get in the numerator AX_0 plus BY_0 plus CZ_0 plus D and in the denominator we have AX_d plus BY_d plus CZ_d. Further simplification can be observed by looking at this as nothing but in the dot product form of vectors. This is nothing but the dot product of the normal with R_0 and this is nothing but dot product of normal with ray direction.

(Refer Slide Time: 00:19:36)



Before you explicitly compute t you probably need some word of caution. You do not want this to go to 0 so if this is 0 what does it indicate? It indicates that the ray is parallel to the plane and therefore there is no intersection.

(Refer Slide Time: 00:22:37)

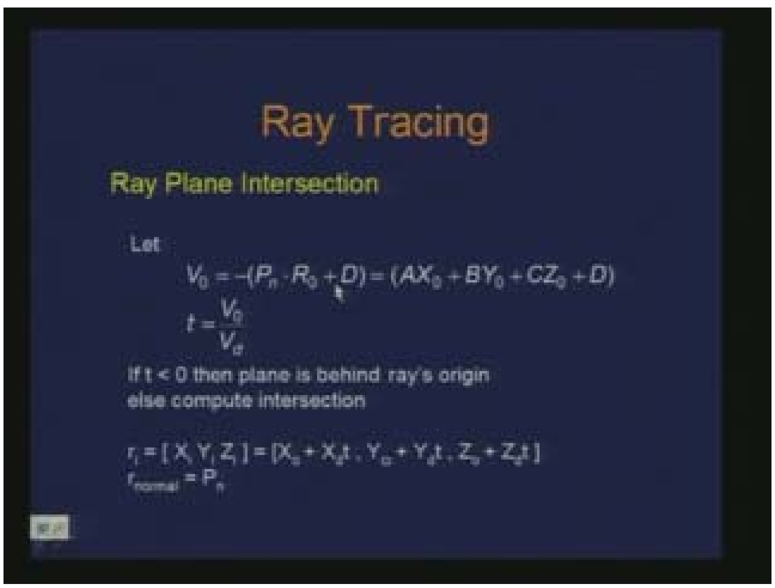


If I just look at the denominator part of it defined as V_d which is the dot product of the normal and the ray direction and then if I check for whether it is 0 or not then only I proceed further intersection. So if it is 0 there is no intersection. There is also another condition which is useful that this is the ray direction and this is the plane which I am considering and this is the normal to the plane.

Now this V which is nothing but dot product of the normal with the ray direction if it turns out to be greater than 0 what does it indicate? In that case this normal would have been in this direction so the plane is facing the opposite direction of the ray. Therefore with respect to the visibility if the ray is considered as the ray coming from the viewer and if there is a normal which is pointing away from that you may declare that it is not visible. But if it is a different situation that you need to assign certain attributes of color or illumination a light ray or another ray which we use for painting the plane then I am talking about the other side of the plane.

Therefore, basically I can consider the plane to be two sided entity there could be front side and back side which may have its own illumination. Then I can use the flip version of the normal to compute the illumination at that point. But here if there are only one sided planes we are not interested in rendering the back side of the plane so we just eliminate that. This is how we compute the intersection. As far as the numerator part is concerned that is what we had in the numerator which I can call as V_0 so it is just t is equal to V_0 by V_d.

(Refer Slide Time: 00:26:55)



And again if t turns to be less than 0 then plane is behind rays origin therefore do not compute intersection. This will be elaborated further because we have already computed t which in turn gives you an intersection point and there could be further utility of this. And if we find that t is in the valid range then we substitute that t to get the point of intersection which is X_i Y_i Z_i which is this and the normal is nothing but P_n itself.

Now the problem is that once we have figured out the intersection with respect to a plane defined in its implicit form A_x plus B_y plus C_z plus D is equal to 0 typically you would have an object defined as a plane which has boundaries. This is sort of an infinite plane so your object would have certain extends or you may have an object which is composed of several polygons or collection of polygons then this becomes only part of the intersection computation. So you need to establish whether the point which you have obtained here through this intersection is within the extends which you have defined for the plane. That is what we call as the containment test of the point whether that point is contained within the extends of the polygon. How do we check that? The polygon is actually defined through the boundary points of the polygon and now you have also found out the point of intersection with that plane so this R_i is known to you. The question which is being asked is to answer whether this R_i lies within that boundary points or not. The parameter of the boundary is also possible.

One possibility is scan conversion. Therefore if I find out a point P here and if I shoot a ray in some direction then the number of intersection with the boundaries of the polygon can indicate whether the point is inside or outside. It is the same as this parity check. Again there could be special cases if the point is like this. Hence one way to do is, consider a ray coming from that point in some direction and find out the intersection with the edges of the polygon. And the number of intersections is an indicator to whether the point is inside or outside. If the number is odd then the point is inside and if the number is even then the point is outside.

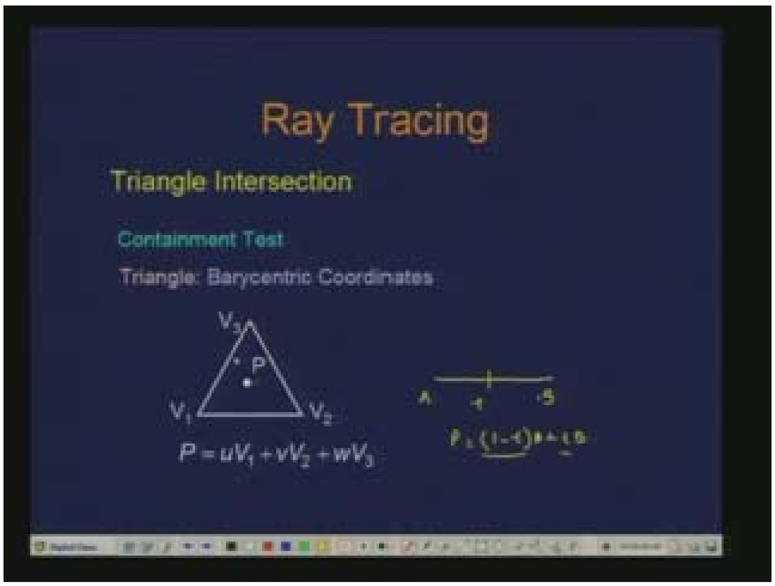
(Refer Slide Time: 00:30:20)

There is another way to look at it which is very commonly used particularly when you have the polygon to be a triangle. That is where one can look at the parametric definition of the point with respect to the boundary of the point.

Typically what is there is that you take the projection of the polygon perpendicular to the normal of the plane and there when you consider the point then you can fix the axis or a direction so you can also compute some sort of a principle direction of the polygon and then start shooting in that direction. Or you can align the polygon in some coordinate access and then shoot the ray in that direction.

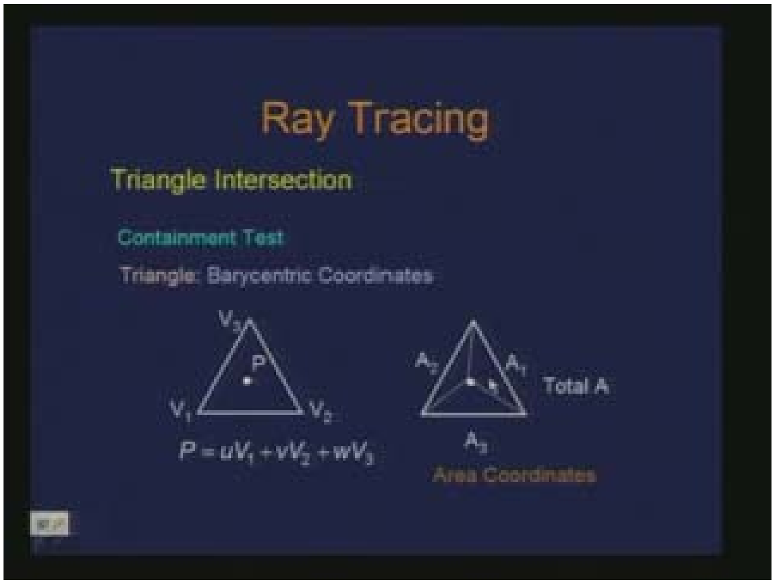
Now let us see if we consider polygons to be the triangle hence we have a better way to see this process of containment. So the question which we are asking here is, first of all given this triangle V_1 V_2 V_3 as its vertices there is a point P here inside this triangle. So we would like to represent this point P in terms of these vertices the vertices of the triangle. And the way it is done is we use what is called as Barycentric coordinates. Barycentric coordinates actually gives you a desirable combination of the parameter U V W or the weights which you can use for the various vertices. Now how we compute these Barycentric coordinates? This is something similar to if you had a line A B. Now for a point within this line

(Refer Slide Time: 00:33:40)



If I have the parameter t then I can always defined this point P as (1 minus t)a plus tB. So, this 1 minus t and t are nothing but the Barycentric coordinates because here we are basically defining a point on the line as a combination of the two end points using a parameter. Or I can write U_a plus V_b where U plus B is equal to 1. This U and V are nothing but Barycentric combination of the Barycentric coordinates for that point. Similarly we are trying to compute these U V W now because we have three points at the boundary which are the Barycentric coordinates.

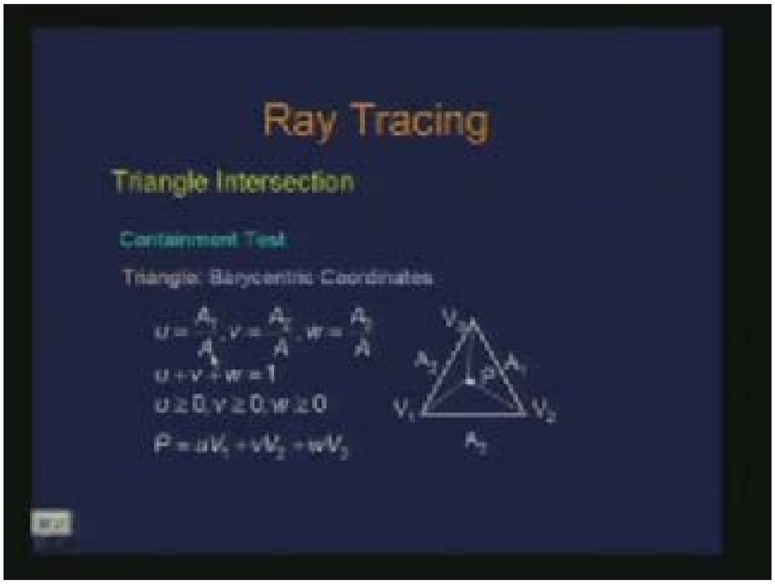
(Refer Slide Time: 00:35:28)



Here we had used the distance, t is nothing but a distance measure from the two end points. It is the ratio of distance. So in the case of triangle we use areas. They are also called area coordinates. If I define this area in front of the vertex V_1 as A_1 similarly A_2 as this area and A_3 as this area so the total area is A of the hole triangle. Then these Barycentric coordinates or the area coordinates are nothing but the ratios of these individual areas to the total area and A_1 is opposite to V_1.

You can also define E_1 the edges also. Either you can take the vertices or the edges. So, if I am defining V_1 as this then E_1 is defined as this edge. So we define these U V W as nothing but the area coordinates which means this U is nothing but A_1 by A, V is nothing but A_2 by A and W is nothing but A_3 by A, and we also observe that U plus V plus W is equal to 1 and all these coordinates are greater than or equal to 1. So they are convex combinations.

(Refer Slide Time: 00:41:55)



So how does it help in containment test?
If there exists U V W we basically have to compute the U V W from the point and figure out whether these conditions are satisfied. If these two conditions are satisfied then the point is inside otherwise the point is outside. Now, where else we can use these Barycentric coordinates?

This is basically doing some sort of an interpolation with respect to the vertices of the triangle. We have looked at the shading models namely polygon shading, Gouraud shading, phong shading and there we basically do this. We do an interpolation of the point to find out the intensity or the normal so the only difference is there it was considered for a general convex polygon and here I am considering the triangle. So in fact if you use this triangle as the polygon some of the limitations or the artifacts which could be produced there can be eliminated because you are always fining the combination with respect to the vertices of the triangle.

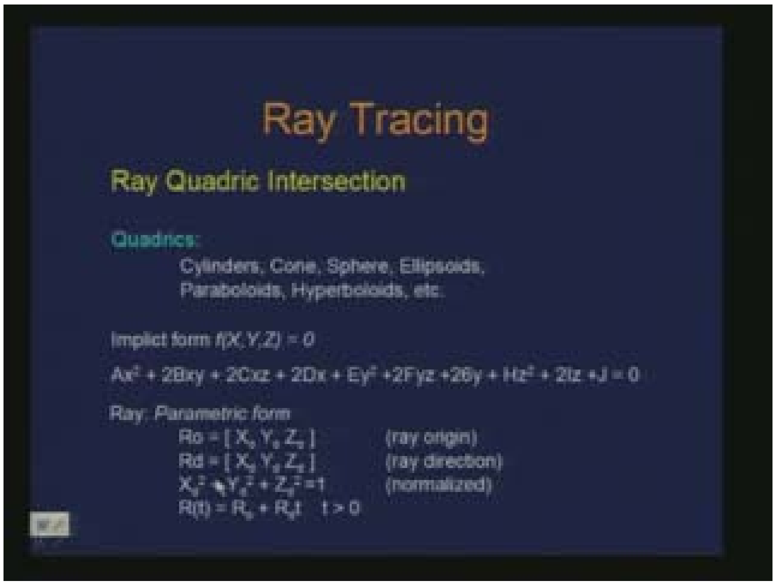
When the rotation of the polygon is done the edges become different therefore the point gets different coordinates for different intensities. So if I do the combination with respect to the vertices of the triangle that will never happen. You can use this at various applications. Basically this takes care of ray plane intersection. These are also unique. There could be a different representation not necessarily the area coordinates.

Convex combination is not this combination, the uniqueness is not preserved. In fact this can be extended in 3D where 3D means if I am having a volumetric element. We started with a line where we looked at the Barycentric combination to get a point on the line. Similarly here the point inside the triangle is a Barycentric combination of the points at the vertices.

Now if I take a volumetric element tetrahedron for instance then again the containment test within that tetrahedron can be established using Barycentric coordinates where the Barycentric coordinates would be computed using the volumes ratio. Therefore there you will be using the volume ratios just exactly the same way as the area ratios. For instance here you will have V_1 the volume of the tetrahedron which would get formed by the point inside be tetrahedron divided by the total volume.

Ray quadratic intersection: Quadrics are shapes such as cylinders, cones, sphere, ellipsoids, paraboloids, hyperboloids etc. So again the idea is very similar to what we have done in the case of sphere or plane so you have the implicit form of a quadratic like f(X, Y, Z) is equal to 0 which is given here A_x power 2 plus $2B_{xy}$ plus $2C_{xz}$ plus $2D_x$ plus E_y power 2 plus $2F_{yz}$ plus $2G_y$ plus H_z power 2 plus $2I_z$ plus J is equal to 0.

(Refer Slide Time: 00:43:13)



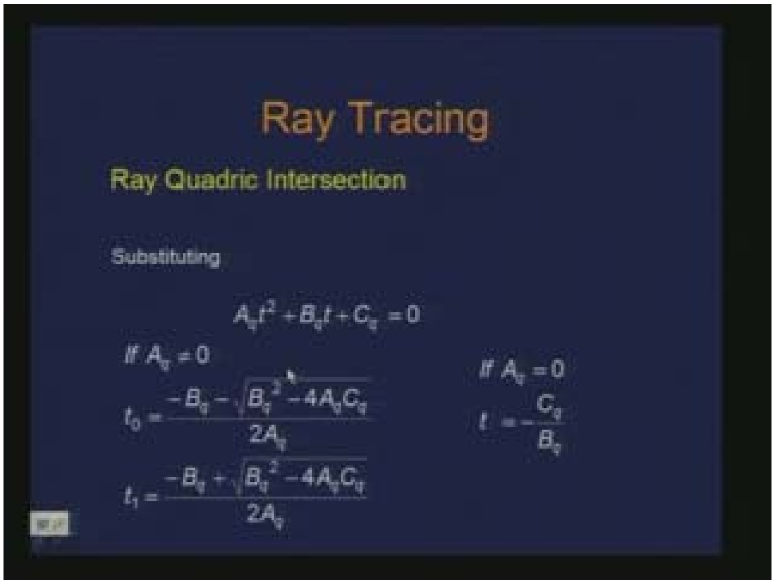
So again if I consider parametric form of the ray then it is just a matter of substituting the equation into this and solve for t. It is exactly the same way as it was done for a sphere.

(Refer Slide Time: 00:44:12)

There is an interesting way of representing the same quadric in a matrix form where this is in the homogenous form [X Y Z 1] then the coefficients are represented by a 4 into 4 matrix which is a symmetric matrix so A B C D and A B C D here in the column so this part of the matrix is the same as this part of the matrix. Again you have the column of the point [X, Y, Z] is equal to 0 so this is also a same quadric which we looked at.

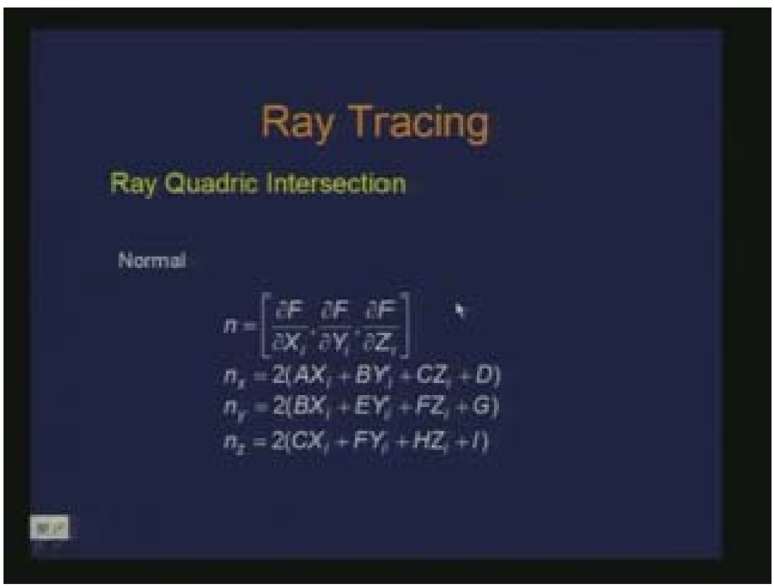
Again we can solve this as a quadric equation in t and if we have the condition that A_q is not equal to 0 we have the two values of t given here with a change sign of this minus here and plus here and if Q happens to be 0 then we have a simpler solution t is equal to minus C_q by B_q. Once we have again found the point of intersection the next thing as what is the normal at the point of intersection. How do we find out the normal? Clearly it is not as simple as it was in the sphere.

(Refer Slide Time: 00:45:02)



So what we have? We have the implicit form f[X, Y, Z] is equal to 0. So how do you find out a normal for f[X, Y, Z]? We take the del function. So the normal is nothing but taking the del function for f which is del F by del X_i del F by del Y_i del F by del Z_i which turns out to be…….46:15. Now the illumination computation can be done.

(Refer Slide Time: 00:46:37)



Ray box:
A box can be thought as a collection of polygons and then you basically perform an intersection with individual polygons of the box and then see what the points of intersection are.

(Refer Slide Time: 00:50:50)

This was a scenario where you could ignore the property of the box as such and just think about this as a collection of polygons and then find out the intersection with each polygon and the polygon with which it intersects you again sort on those to figure out the point of intersection. But an alternate way is the context of clipping. The 3D clipping is nothing but finding out intersection of the line which is to be clipped against the box which is the clipping window.

It is just a matter of applying your Cyrus Beck or Liang Barsky clipping algorithm and it requires you to have this box to be a convex polyhedral. So typically when we are looking at box they are basically linear boxes, parallelepiped structures for which applying this 3D clipping is fairly easy.

If I have X in this direction and Y in this direction then the same treatment can be done if I add Z to it. Then I am talking about the potential points of intersection or the ts corresponding to the potential points of intersection with this ray this is the ray I have as this point so this point is nothing but a near t point or t near point or t corresponding to the near point and this is t corresponding to the far point which are also thought as the entering points and the leaving points when we were doing clipping so this is the same thing.

Similarly, I would have this point and this point as the near and the far point or entering or the leaving point. Again I can figure out this segment which would be inside that volume by clamping the tmin and tmax. So all we have to do is we basically have to just apply our 3D clipping algorithm where ray is nothing but a line through the window of this box.