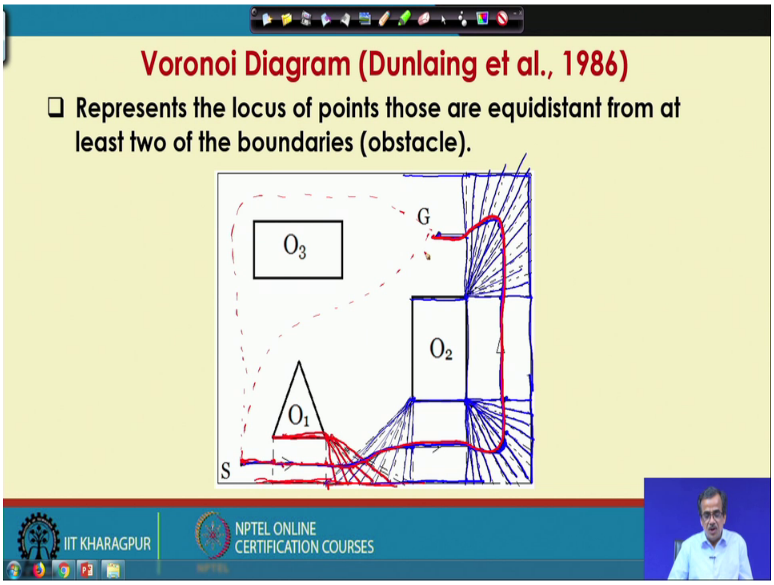
Robotics
Prof. Dilip Kumar Pratihar
Department of Mechanical Engineering
Indian Institute of Technology, Kharagpur

Lecture – 38
Robot Motion Planning (Contd.)

We are discussing how to solve the find path problem using graph based techniques. Now, I am just going to start with the working principle of this Voronoi diagram which is a very popular graph based method to solve the find path problem.

(Refer Slide Time: 00:27)



The concept of Voronoi diagram that was proposed by Dunlaing et al in the year 1986; Now, here the problem is something like this; supposing that this is the field, this is the field and I have got one robots say point robot we have and this is the starting position of the robot. And its goal is denoted by the point G; now starting from here the point S, it will have to reach the point G by avoiding collision with the obstacle.

Now, here we are just going to consider 3 obstacles, 3 static obstacles like O 1, then comes O 2 and O 3. Now, supposing that there is no such obstacle; now if there is no obstacle then it will start from here and it is going to reach this point G following a path something like this, but unfortunately there are some obstacles there are some fixed obstacle. Now, let us see how to find out the collision free path using this particular your the Voronoi diagram.

Now, in a Voronoi diagram actually what we do is; we try to find out the locus of the points which are equidistant from 2 boundaries. Now if I start from the point S and our aim is to reach this particular goal. So, here the most critical obstacle is nothing, but O 1 and we have got the boundary of this particular field and this is nothing, but the boundary of this particular field and at the same time this is nothing, but the boundary of the obstacle.

So, what we do is; we consider so this particular boundary of the obstacle and the boundary of the field and we try to find out the midpoint. So the midpoint of this and this; so is nothing, but this particular this point. So, this point is equidistant distance from this obstacle boundary and the boundary of the field. Similarly, I can find out the midpoint of these 2 boundaries are something like this and I will be able to reach this particular the point.

And once I have reached this particular point; now I will have to consider, so, this particular point that is the vertex of this particular obstacle and the boundary of this the field. So, if I consider this is one point and this is another point lying on the boundary; so, this is the midpoint. Similarly we are going to consider a few more distances from this particular point and we try to find out what should be the midpoint and what should be the locus of the midpoint.

For example on this particular straight line the midpoint could be here; here the midpoint could be here, the midpoint could be here, the midpoint could be here and here the midpoint could be here. So, what you do is we start from here and then up to this we can find out your, the collision free path; the collision free path could be something like this. For example, starting from here the collision free path up to this it is something like this, then I can find out a path up to this.

Next we try to consider; so, this particular vertex of the obstacle that is O 2 and this is the boundary of this particular field. So, what you do is we try to draw some straight line; so, for example, say from this particular boundary to this point; so, these are the straight lines we consider. So, these are the straight lines we consider and try to find out the midpoint; so, the midpoint of this particular straight line is this, here this is the midpoint, this is the midpoint, midpoint and we try to join by a smooth curve and this could be the path.

So, we have reached up to this; that means, the point robot has reached up to this; now we consider so this particular the boundary and the boundary of this particular field and the locus of the midpoint starting from here; so, this will be the locus up to this. Now after that we consider, so this particular vertex of O 2 and this is the boundary; so, we draw the straight lines, we draw all such straight lines here. And for each of this particular straight line we try to find out the midpoint. So, these are the midpoints; so, from here we can we can just join by a smooth curve. So, the point robot has reached up to this.

Next we consider; so this vertex and this is nothing, but the boundary of the field. So, this is the straight line; so, these are the straight lines we consider. And we consider the midpoint of this particular straight line; so, from here there is a possibility that I will be getting this type of the path. So, the point robot is here; so from here to here; so this is the boundary of the obstacle and this is the boundary of the field. So, it is a midpoint the locus of the midpoint could be something like this; so, from here I will be able to reach this particular the point.

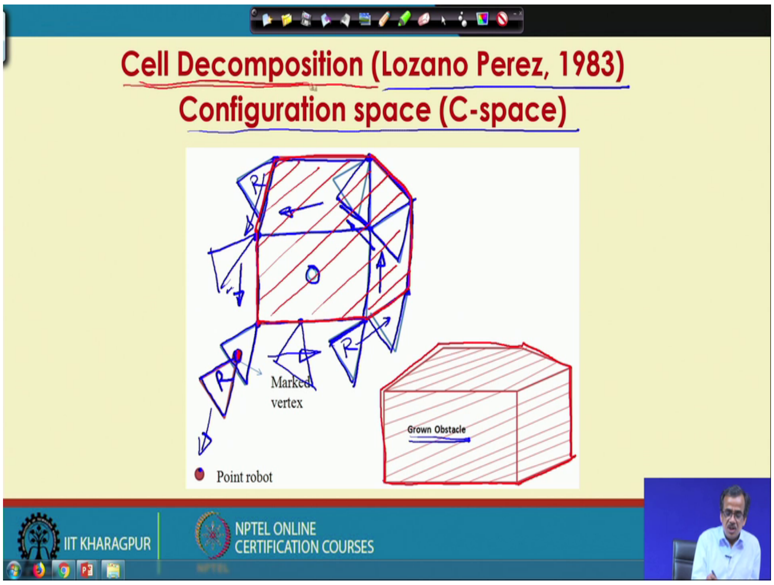
Next we consider this vertex and this is the boundary of this particular field; so, we join by straight line. So, these are the straight lines and we try to find out the midpoint; so these are the midpoint and we join by a smooth curve. So, up to this; so this is the path now we consider, so this vertex and this is the boundary of the field. So, these are the straight lines; so these are the straight lines which you are going to consider and once again we try to find out the locus of the midpoint. So, it will be getting say this type of locus and then from here to this particular goal; so, this is the obstacle boundary and this is the boundary of the field. So, from here the locus will be something like this.

So, starting from here; starting from your the initial point. So, this was the initial point; so, I can find out a collision free path which is something like this. So, this is a collision free path for this particular the point robot; so this is a collision free path. Similarly, so we can start from here and I can move along this particular direction and there is a possibility that I will be able to find out another feasible path something like this or I can I will be able to find out another path something like this.

So, there is a possibility that we can find out several such feasible paths and out of all such feasible paths actually we will have to find out the time optimal path. But in a

Voronoi diagram actually they did not try to find out the time optimal path; they only tried to find out the obstacle free the collision free path. So, this is one of the possible the collision free path for the, the point robot. So, this is the way actually using the principle of Voronoi diagram; so, we can find out the collision free path for the point robot in the presence of some static obstacle. That means, we can solve the find path problem using the principle of the Voronoi diagram.

(Refer Slide Time: 08:48)



Now the next is your; the concept of cell decomposition. Now this particular cell decomposition is another very popular the graph base technique to solve the find path problem. The concept was proposed in the year by Lozano Perez in the year 1983, so it was proposed in the year 1983 and Lozano Perez; he proposed one technique that is called the cell decomposition and he also gave the concept of so this particular the configuration space or the C space.

Now let us try to see the problem; now here actually what we do is we consider one physical robot in place of point robot and we consider one static obstacle. Supposing that the physical robot is something like this, so this is the physical robot which I am going to consider and this is the marked vertex of this particular the physical robot. And supposing that I have got one static obstacle and the boundary of the obstacle is something like this. So, this is nothing, but the obstacle; so this is nothing, but the obstacle and I have got a physical robot something like this. So, this is the robot and we have got the obstacle here; how to ensure the collision free path for this type of the robot ok?

Now, what you do is; so this particular physical robot that is converted into a point robot something like this and this obstacle. So this type of rectangular obstacle that is converted into; so, this type of grown obstacles something like this. Now, how to reach this grown obstacle starting from this particular obstacle? That I am going to discuss. So how to, how to replace this physical robot by point robot and how to replace so this particular obstacle; by a modified grown obstacle so that the problem remains the same.

Now, what is the problem the problem is we will have to find out a collision free path for this particular robot so that it does not collide with this particular the obstacle. And this problem is equivalent to determining a collision free path for the point robot in the presence of; so this type of the grown obstacle.

Now, let us see how to how to achieve this grown obstacle; now here I am just going to put one condition, the condition is as follows. So, this particular marked vertex; so its orientation will remain the same. So, what you do is. So, I have got this particular obstacle and this particular the robot I just place it here. The robot is placed here and this particular marked vertex is going to coincide with this particular the corner of the obstacle.

Now, actually what you do is we try to slide in this particular direction; so, the robot is going to slide in this particular direction. So, there is a possibility this will be the position and once again it is sliding and then after sometime, it will reach this particular point. So, this is the marked vertex and once it is reached this what we do is the this particular robot R can slide along this particular your the edge. So, what will be getting is your; so, it can slide in this particular direction.

So, this will be the marked vertex and this will be the position of the robot and once it is reached this now it can slide in this particular direction keeping this particular marked vertex, the orientation of the marked vertex is the same. So, I can slide in this particular direction then gradually I am just going to reach this particular the position. So, this will be actually the locus of the marked vertex.

Then here actually what I can do is I can slide in this particular direction; so, there could be sliding here and there is a possibility if it slides it will take the position something like this. And this is the orientation of the marked vertex and this will be the locus of this particular the marked vertex. So, after that; so it can slide in this particular direction, then gradually I am just going to reach this particular point. So, the marked vertex will be here and this will be the position of this particular; the robot R.

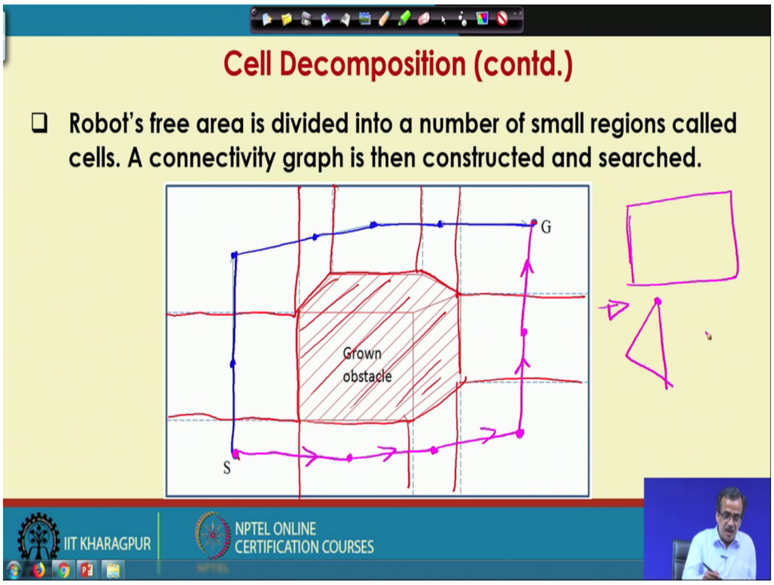
So, this will be the locus of this particular the marked vertex and after that; so, here it is going to slide in this particular direction. So, there is a possibility that your this will be the situation. So, this could be the situation and this is the marked vertex. So, the marked vertex is sliding along this and then the marked vertex is going to reach this particular the starting point and this is the direction the robot is going to slide.

Now, if this is the situation then very easily we can find out the locus of this particular the marked vertex. For example, we started from here then the marked vertex moves like this then it moves like this, then it moves like this, then it moves like this, then it comes here, then it will come here and then it will come here. And this is nothing, but is actually the grown obstacle.

So, this particular robot will be replaced by the marked vertex; so, this particular marked vertex is nothing, but a point. So, this is the point and the original obstacle; so, that will be converted to the your the grown obstacle. So, this particular grown obstacle I have just now redrawn it here; so, this is nothing, but the grown obstacle.

So, now the problem is equivalent to a point robot and this particular the grown obstacle. So, I will have to find out the collision free path for this particular point robot by considering; so this type of grown static obstacle. Now let us see how to tackle this particular problem using the principle of your the cell decomposition.

(Refer Slide Time: 15:49)



Now, here actually what we do is; the same grown obstacle I just re do it here and this is nothing, but the position of this particular the point obstacle. So, this is the position of the point obstacle ok; so this is the starting position and the goal could be here that is denoted by G and this is nothing, but the grown obstacle. So, the grown obstacle is something like this; so, this particular the grown obstacle represent the infeasible zone. The points robot should not reach here just to avoid that particular the collision with the static obstacle.

Now, if this is infeasible zone the rest of the zone will be the feasible zone. So, this is actually the boundary of the field. So, this particular white portion will be nothing, but the feasible zone; now this particular feasible zone this is divided into a large number of small small sub regions. For example, starting from here; so what I can do is, so this feasible zone I can just divide something like this.

So, this is one feasible sub region, similarly this is another feasible sub region, this is another feasible sub region, another feasible sub region, another feasible sub region. So, we will be getting some feasible sub region something like this and once we have got this particular the feasible sub regions. So, what I do is we start from here, go to the nearest feasible sub region; so, this could be the nearest feasible sub region. So, we try to find out the midpoint of this particular the sub region; the center of this particular the sub region.

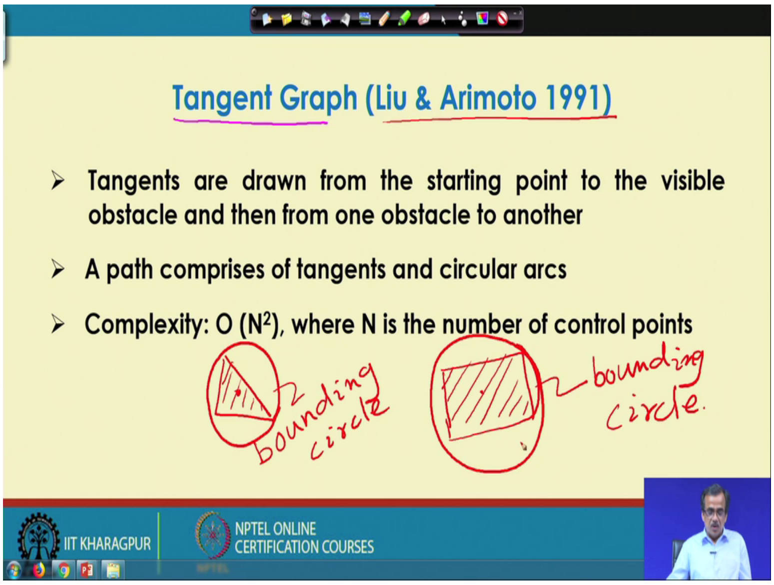
Similarly, this is the center of this particular sub region, this could be the center of this particular sub region, this is the center of the sub region, center of the sub region. Then the path could be something like this start from here, then you reach this particular point, then you come here, then you come here, you come here and the robot is going to reach that particular the goal.

Now, this is one of the feasible paths; there could be some other feasible paths for example, another feasible path could be something like this. You just start from here then you find out this as the sub region; the feasible sub region. So, the midpoint could be here now next is this is the feasible sub region the your midpoint could be here, the center of this particular sub region could be here, this could be the sub region; the center for the sub region.

So, another feasible path could be something like this; so, this could be another feasible path for this particular the point robot. So, this is the way using the principle of the cell decomposition method; so we can find out the feasible collision free path for the the robot. Now, here we are trying to find out the collision free path for the point robot considering this grown obstacle.

Now, by solving this we are also able to solve the collision free path for the actual physical robot which are something like this and this was the marked vertex and original obstacle was something like this. So, if you solve this particular find path problem; so, indirectly we are solving; so this particular the find path problem. So, this is the way like using the principle of your the cell decomposition method; so, we can find out the collision free path.

(Refer Slide Time: 19:42)



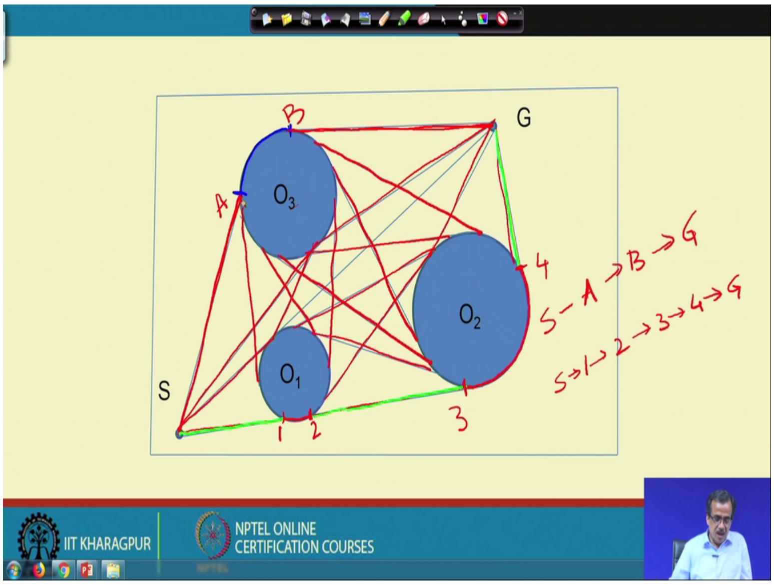
Now we are going to start with another very popular method; now in very popular method for solving the find path problem. And this is known as the tangent graph technique, now this tangent graph technique it was proposed by Liu and Arimoto in the year 1991. Now here actually what you do is we try to move along the tangent of this particular the circles.

Now, supposing that I have got; so this type of say the fixed obstacles, a triangular fixed obstacle or there could be some sort of say rectangular fix obstacle sort of thing. So, here actually what we do is we try to draw one bounding circle for these particular the obstacle. Now supposing that this is actually the triangular obstacle; so we try to find out the center and considering these as the center. So, we try to draw one circle and this particular circle will be the; the bounding circle for this static obstacle that is the triangular obstacle.

So, we try to find out the collision free path considering; so, this particular the circular boundary and this is nothing, but the bounding circle. So, this is the bounding circle for this particular your the static obstacle. Similarly if this is the static obstacle, we try to find out the center of the area and once again we try to draw one circle. And this is will be nothing, but the bounding circle for this particular your the static obstacle.

And once you have got this particular bounding circle; so, instead of considering the physical dimension of these particular obstacle; So, we are going to consider; so, this type of boundary the bounding circle. Now, here; so let us try to find out like the feasible path for a point robot considering so this particular the bounding circle. So, once again let me repeat like for each of this particular static obstacle, we try to find out the bounding circle. And considering the bounding circle we try to determine what should be the collision free path for the point robot?

(Refer Slide Time: 22:14)



Now, here if you see say this is the find path problem say we consider this is the initial position; that starting position of the point robot and this is the goal. Now, if there is no such obstacle; so very easily starting from here, so it is going to reach that particular your; the goal. And absolutely there is no problem because there is no such obstacle, but if I consider the obstacle here. So, the path will be slightly different; how to find out that particular path? To find out the path we take the help of the tangent graph technique; the technique is as follows, we start from this particular point S and we try to find out which one is the most critical obstacle.

Now, if I compare O 1, O 2 and O 3 these are all static obstacle; so out of these 3 obstacles, so O 1 is the physically the closest. So, we first consider; so this particular O 1 obstacle. So, what I do is; so from here we try to draw the tangent, so from here we can draw one tangent which is something like this. From here we can draw another tangent to this particular circle; then we consider; so this particular obstacle and the obstacle that is O 3.

And we try to draw all the external and the internal tangents for example, one tangent could be something like this, another tangent could be something like this and we can also consider; so, this type of tangents. So, these are also tangents ok; next we try to find out the tangent between so this particular obstacle and this particular obstacle. But before that starting from S, I can also draw this particular tangent I can draw another tangent here ok. And between these 2 obstacle; so I can find out the tangent something like this. So, this is one tangent, this is another tangent, then we get another tangent here; we get another tangent here.

Now, from here this obstacle O 2; so, I can find out; so this type of tangent also between O 1 and O 2 then this is another tangent ok. Then I can find out another tangent another tangent ok, I can draw the tangent here I can draw the tangent here. Similarly this is another and I can draw this type of tangent also.

Now, once you have got these particular tangents; now we are trying to find out what should be a feasible path. Let us see how to find out a feasible path; now to find out the feasible path actually what you do is supposing that I am just going to start and this is the first critical obstacle. So, let me let me follow this particular path ok; so from here supposing that I am just going to follow this particular tangent. The next tangent is here; so from here, so the next tangent could be here up to this. And this point obstacle will follow this circular path and it will reach this, then it is going to follow this particular the tangent; so, this is one feasible path.

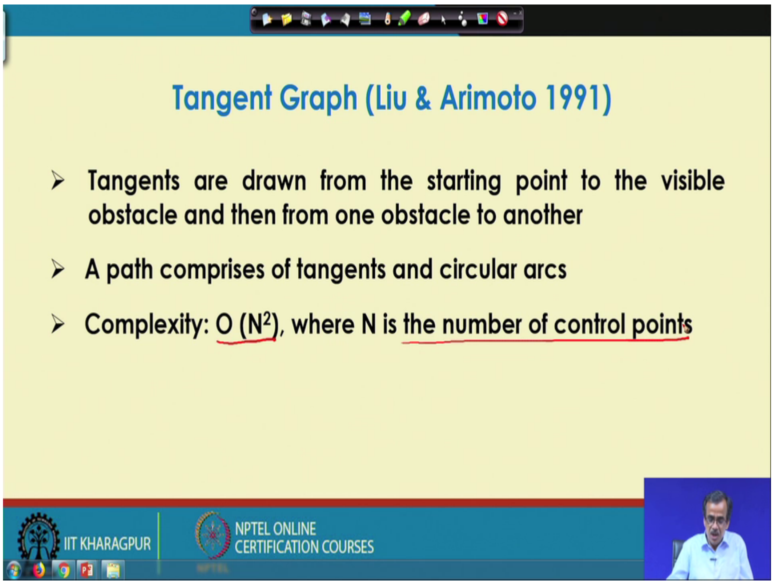
Now, here in this particular feasible; path we can see that like we can draw; so, some circular arc here for example, say from here to here, there will be a circular arc. So, from here to here there could be another circular arc something like this. Now if I draw one circular arc here another circular arc here; supposing that this is denoted by point 1, this is denoted by point 2 ok, this is denoted by point 3, this is denoted by point 4. So, the feasible path will be your S to 1, 1 to 2, 2 to 3, 3 to 4, 4 to G; so, this is one feasible path.

Similarly, there could be some other feasible path for example, say I can also consider some other feasible paths. For example, another feasible path could be something like this; so you start from, here then you reach this particular point, then comes your. So, from here; so you follow these particular circular paths up to this.

Now from here actually what you can do is; so you can follow; so this type of the tangent path ok. So, if I just write down say this is A, this is B and this is G; the path could be your S to A, A to B, B to G. Similarly, we can find out the sable such feasible paths and out of all the feasible paths; in fact, we will have to find out what should be the most the time optimal and collision free path.

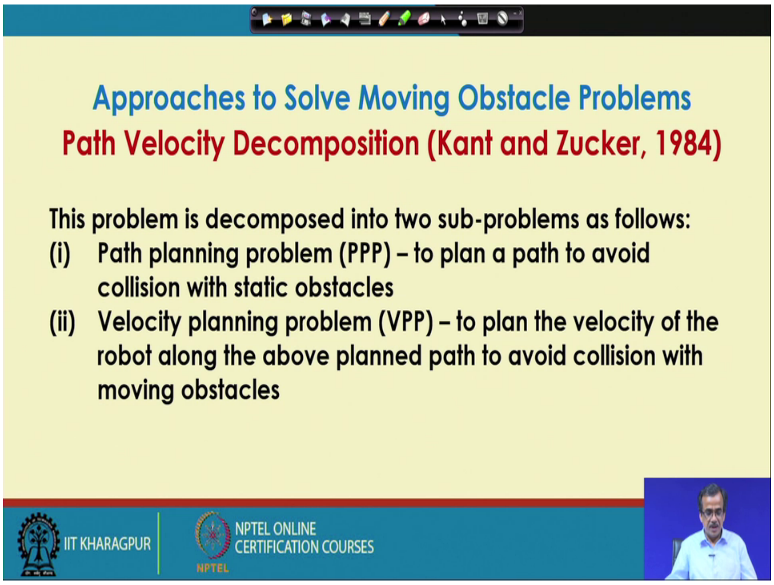
Now, this particular algorithm actually could reach some popularity and as it is using the principle of tangent. So, there is a possibility that you will be getting the optimal path, but here the main problem is actually the computational complexity. Now, if you see the computational complexity it has been checked that the computational complexity of this particular algorithm is nothing, but order of N square.

(Refer Slide Time: 28:12)



Or N indicates the number of control points or the number of circular arcs ok. So, the more the number of control points or the circular arcs, the more will be the complexity. And the complexity is actually quadratic it is the square, square is the order of N square; N is nothing, but the number of circular arcs. Otherwise this method is very good and it could solve the find path problem very efficiently. So, this particular the method as I told could reach good popularity.

(Refer Slide Time: 28:52)



Thank you.