Robot Motion Planning

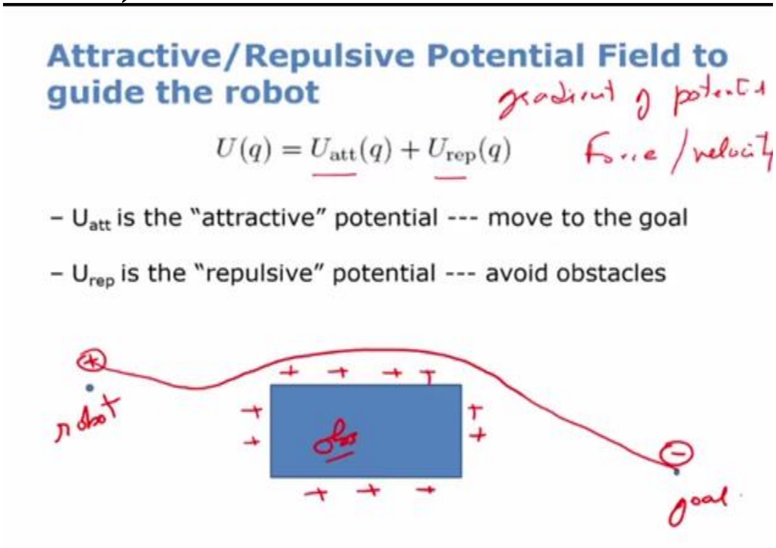
Prof. Ashish Dutta
Department of Mechanical Engineering
Indian Institute of Technology – Kanpur

Lecture – 14
Navigation Function and Potential Field in 3D

Hello and welcome to lecture number 14 of the course robot motion planning. In the last class, we looked at the potential field method. And we will continue with that discussion today then we will look at path planning using a navigation function. And then from there we will move on to potential field method in 3D. So, if you remember whatever we are talking about in the potential field method, till now is basically limited to 2D cases.

That means you have flat ground in which you have obstacles and the robot basically avoids the obstacles and goes to the goal point. So, from there, we will continue with this discussion, and then move on to navigation function. And then we look at potential field in 3D, very briefly revising what we had done in the last class regarding the potential field method.

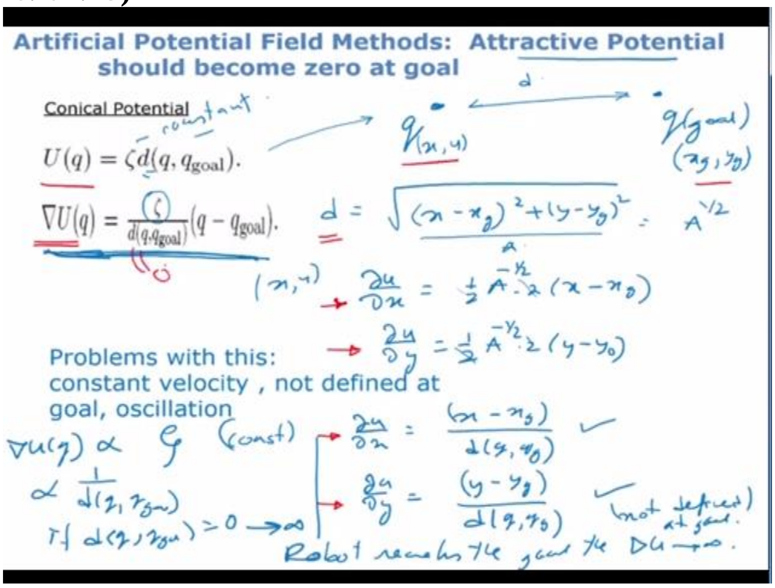
(Refer Slide Time: 01:01)



So, in the potential field method we basically assign potentials attractive and repulsive potentials to the goal, the robot and the obstacle. So, if you have a robot here and we have an obstacle here and my goal is given here. So, we basically assign attractive and repulsive potentials to guide the robot avoiding obstacles to reach the goal. For example if I assign plus, plus charges here and I also assign plus charge to the robot then obviously so there is plus charge here so the robot is going to be repelled by the obstacle.

And we give a minus charge to the goal so the robot essentially will move like this, you will avoid the obstacles and then you go to the goal. So, basically we avoided attractive potential and repulsive potential and at every point in the vector field the robot is subjected to a force which is the gradient of the differential. So, the gradient of the potential is a force with the robots towards the goal avoid obstacles only for velocity we can also say it is the velocity of the robot. So, now we have to assign attractive potential that means we have to define a function attractive potential function and define also a repulsive potential function.

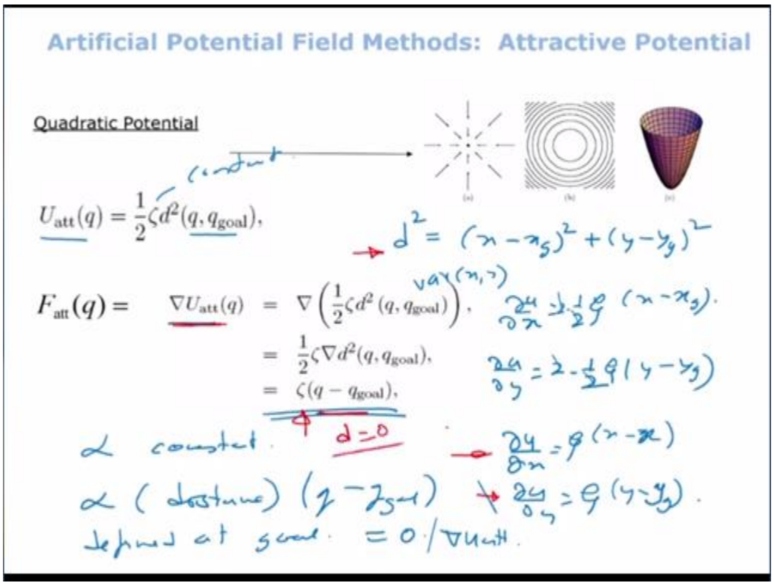
(Refer Slide Time: 02:19)



So, what we saw in the last class is that the attractive potential can be a conical potential which is equal to U which is, $U(q) = \zeta d(q, q_{goal})$. So, as shown here if $q_{(x,\ y)}$ is present position of the robot, and $(x_g\ , y_g)$ is the goal position of the robot, then d is the distance between the goal and the present position of the robot and if you write this as d, then the distance is given by $d = \sqrt{\left(x - x_g\right)^2 + \left(y - y_g\right)^2}$.

When I differentiate this function u with respect to x, I get this term and if I differentiate this with respect to y partial differentiation with respect to y I get this term. So $\frac{\partial u}{\partial x}$ and $\frac{\partial u}{\partial y}$ y are given by these 2 functions and from here is what we get the gradient as $\frac{\zeta}{d}\left(q - q_g\right)$. Now, the disadvantage of this is that when the robot reaches the goal, then what would happen this d will become equal to 0 so this gradient will become undefined so that is the problem here.

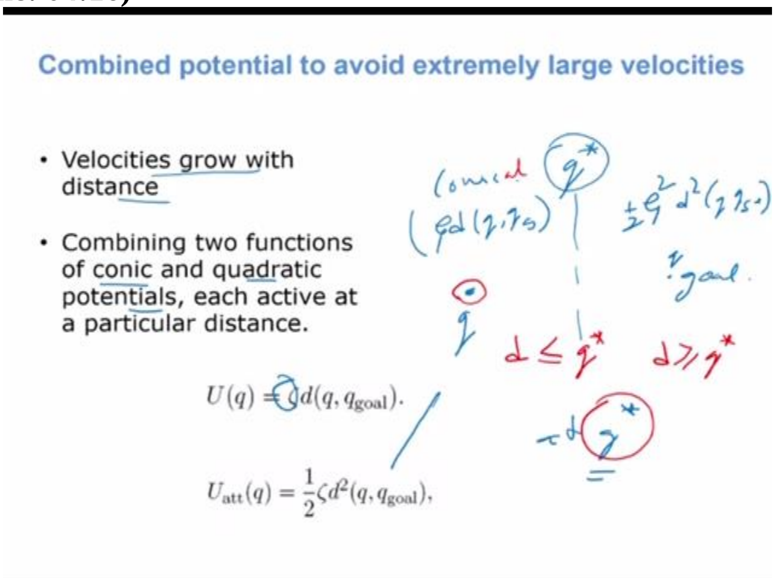
(Refer Slide Time: 03:27)



Now to avoid the problem, what we do is we can use a quadratic potential? The quadratic potential essentially we write in the form $U_{att}(q) = \frac{1}{2}\zeta d^2(q, q_{goal})$, where d is again the distance between the present position in the goal position of the robot. Now, when I take the grad of this, then what we do get this is $\zeta(q - q_{goal})$, exactly the same way distance $d^2 = (x - x_g)^2 + (y - y_g)^2$ and we take the partial differentiation derivative of u with respect to x and then with respect to y, so, I get this term and this term.

So, $\nabla U_{att}(q) = \zeta(q - q_{goal})$ so, when the robot reaches the goal point what would happen is this distance will become d = 0 and the grad will become 0 but it does not become undefined.

(Refer Slide Time: 04:16)

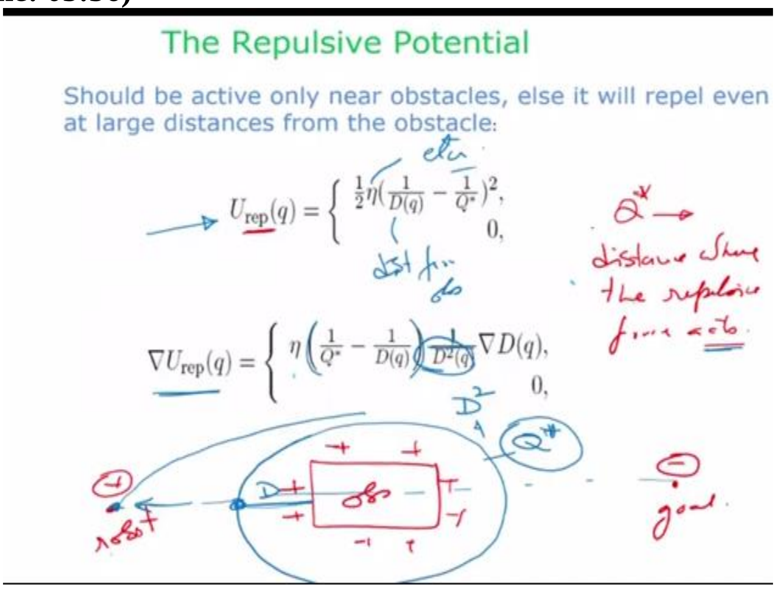


So, what we can do is that we can use both of them the conical and the quadratic potential functions using each of them at a particular distance. For example, we define a distance q * and when the robot is at let us say at point q and the distance is d which is the distance between the goal and the person position d ≤ q * then we say we have to use the conical function and when the robot is d > q *.

Then we say we going to use the quadratic, we can define the values q and q* as whatever we require. Now, something you note here before I proceed that there is a constant which is ζ. So, that means the force acting on the robot, which is the potential will be directly proportional to the ζ. So, how do you get ζ that is a question that might come to your mind? So, ζ is a constant.

So, different numerical values can be assigned for example a high value of ζ will give you a high force, low values it will give you a low force. So, now ζ depends on the person who is writing the program and we will see how ζ is also found, now so this is the conical potential and the quadratic potential functions that are used depending on where is d, d ≤ q * or d >q *. So, that is basically what we call is the effective distance, so let me call this a particular distance.

(Refer Slide Time: 05:56)



Now so much for the attractive potential now we need to define the repulsive potential, repulsive potential $U_{rep}(q) = \frac{1}{2}\eta\left(\frac{1}{D(q)} - \frac{1}{Q^*}\right)^2$. So, where Q * is that Q effective distance. So, Q * is the distance where the repulsive force becomes effective force acts in why Q * is put, essentially, because if you have an obstacle here let us call this an obstacle and you have the robot at this point, and you have the goal at this point. Now, as we said that we have plus, plus here and we also have a plus a minus here and a plus here. So, what will happen is at every point in the vector field?

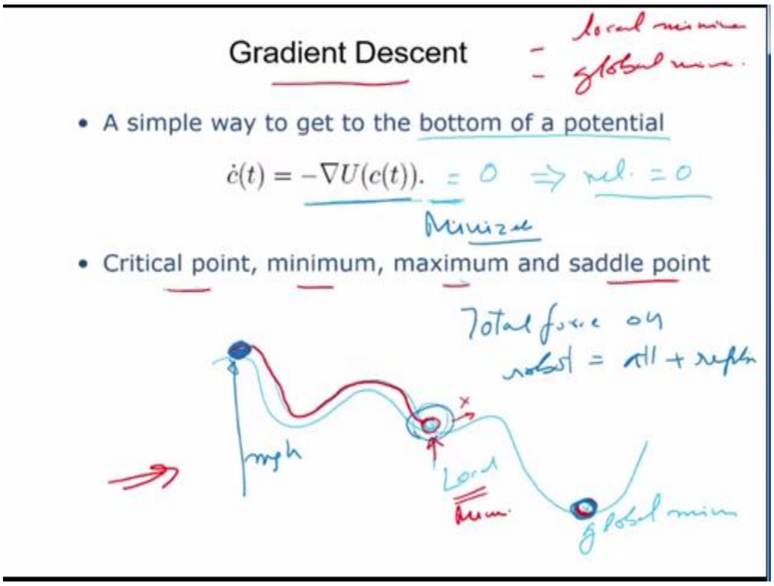
If the repulsive force is also acting on the robot, then the path is going to become longer that is just common sense. So, for example, if we are at this position here, the shortest distance would be a straight line. But if the robot is here and repulsive force is also acting at that point, then it will tend to move the robot like this, whereas the robot could have simply gone like this, like this, like this and then going like that. So, we basically put a distance where the repulsive force starts acting.

So, we put some barrier and say that my Q * is only here. So, this is my Q * which is the distance from the obstacle and d is the distance from the obstacle, this is distance from obstacle. So, if I take the grad of this, what I get $\nabla U_{rep}(q) = \eta \left(\frac{1}{Q^*} - \frac{1}{D(q)} \right) \frac{1}{D^2(q)} \nabla D(q)$e. So, please note that there is a D^2 there, so basically means that D is the distance between the robot and the obstacle.

So, when the robot comes into this distance D, so what it basically means, this is my distance D, what it basically means the force is going to increase very, very quickly because it is a D^2. And at the surface of the obstacle when the obstacle just goes hits that D becomes 0 so this becomes infinity now. So, what you can see is that the way the repulsive potential has been formulated, essentially to ensure that when the robot comes right on the obstacle, the force becomes infinity so that cannot happen.

So, the object or the robot will get pushed off, so this is basically to show that the effective distance Q* actually controls when the repulsive force will become effective on the robot. Now again we have a constant here, this is an η constant so again if η is high repulsive force will be high, if η is lower than the repulsive force will be low. So, we have 2 constants, the ζ, η and we have to find values of ζ and η.

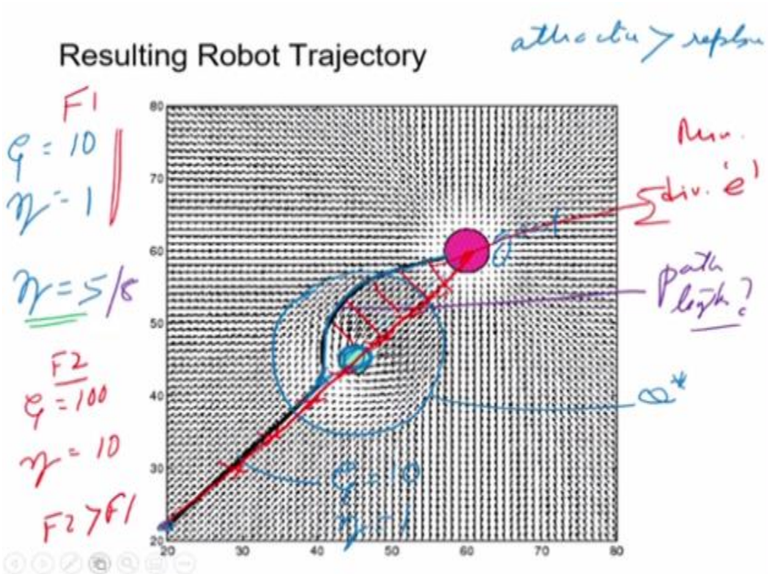
(Refer Slide Time: 08:54)



So, now let us look at how to use the gradient descent method. So, we take the potential, the effective force acting on the robot, so total force on robot is equal to repulse attractive plus repulsive force. So, I use the gradient descent that means I do minimize, so it is a minimization problem. So, basically, what it means is that minimizing the potential energy. So, you can imagine if there is a ball that it has a potential energy mgh when the ball comes down, this decreases.

Now, the problem with this potential field method is that there is a possibility that the robot can get caught in what we call local minima. Suppose we have global minima here, the robot can get caught into local minima's. And this is a problem which is faced by the potential field method. So, what we do here is we use the force acting on the robot and then drive the robot in a direction which reduces the potential by using the negative gradient descent and then we have to be careful that it does not got caught in local minima.

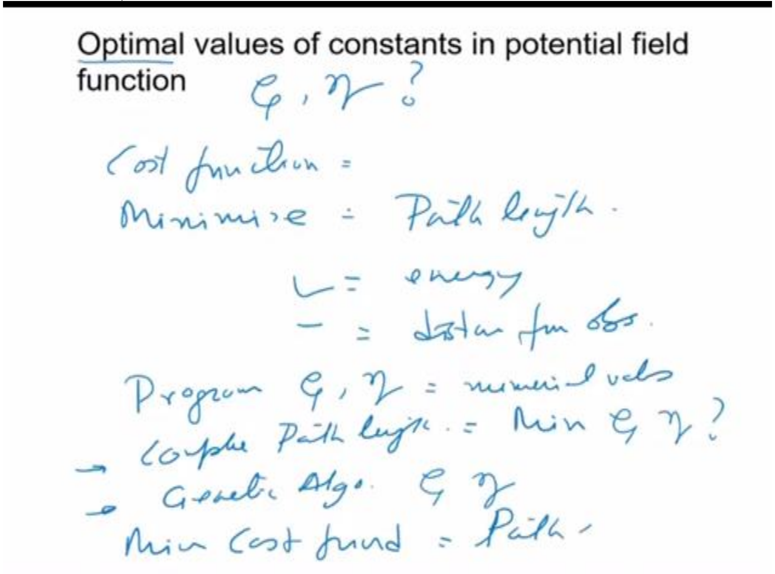
(Refer Slide Time: 10:10)



Now, this is basically a pictorial view of how this whole thing works so, we have an obstacle which is given here and we have a goal which is there and the robot is somewhere here. So, suppose I say my zone of influence is somewhere here. So, this what we are talking about the Q *, this is also called the zone of influence, so only in that distance the effect of the effective force is going to start working not otherwise. So, suppose I have my zone of influence here which is my Q * the robot is coming towards the goal.

Now, it is moving almost in a straight line towards the goal because that is the shortest path, only when it enters the zone of influence then the repulsive force starts pushing the robot away not before that. So, from here the robot is getting pushed away it goes like this, like this, like this and goes to the goal. So, there are 2 constants that we are seeing one is an η and there is ζ, η 2 constants are there.

(Refer Slide Time: 11:12)



So, how do we find optimal values of this now η, ζ. Now how do you get their values there is no so when we use what optimal values, then what do we mean by optimal here. So, optimal could mean first of all whether it has reached the goal that it is a global minima that is one now the optimal values could be with respect to some cost function. Now, what is this cost function or what are we trying to minimize?

So, we are trying to minimize for example, the path length obviously, we want the robot to reach the fastest we can minimize energy of the system; we can minimize distance from obstacle. So, these all could be cost functions that we are trying to minimize when using potential methods. So, what we basically do you can write a program which varies η and ζ put different numerical values compute path length or maybe energy or maybe distance obstacle and then see which is the minimum case.

So, which is the minimum η, ζ, you can use the standard optimization like genetic algorithms. So, then genetic algorithms can vary η and ζ and basically, we can have a cost function which is going to minimize. So, minimize cost function which can be the path length, so what this fundamentally does? If I were to show you pictorially here, suppose we start saying $\zeta = 10$ and $\eta = 1$, now something to remember is that the attractive has to be greater than the repulsive.

Otherwise the robot whenever is the goal is repulsive forces more than to tend to push the robot really away and it may never reach the goal to attractive is normally higher than repulsive and there is a reason I put $\zeta = 10$ and $\eta = 1$. And suppose we are getting this path for $\zeta = 10$ and $\eta = 1$, now suppose I increase η suppose $\eta = 2$ then what do you expect as $\eta = 5$ for example, what do you expect? You expect that the robot will come like this I go further away now like that, why because this is become higher now.

Now, if I make it even higher, then what will happen is suppose I make a 10 and then let me make this 8 just for example, some of the colour let us say this is 8 now, what will happen is it will go like this as soon as it is in the zone of influence will get pushed out. So, what you are seeing is here is that the path lengths are different. So, now we can say that put a value of η and ζ such that we get the best path length shortest path best path length would mean the shortest path length.

Now, for example, I say the path length which will be the how to define path length or which is the shortest path length suppose you are writing a program and you want to minimize the path length minimize this. So, what is the shortest path length? The shortest path length towards the straight line between the goal and the length. So, if I come back here, and I say that I am joining this with a straight line, now that is my shortest length.

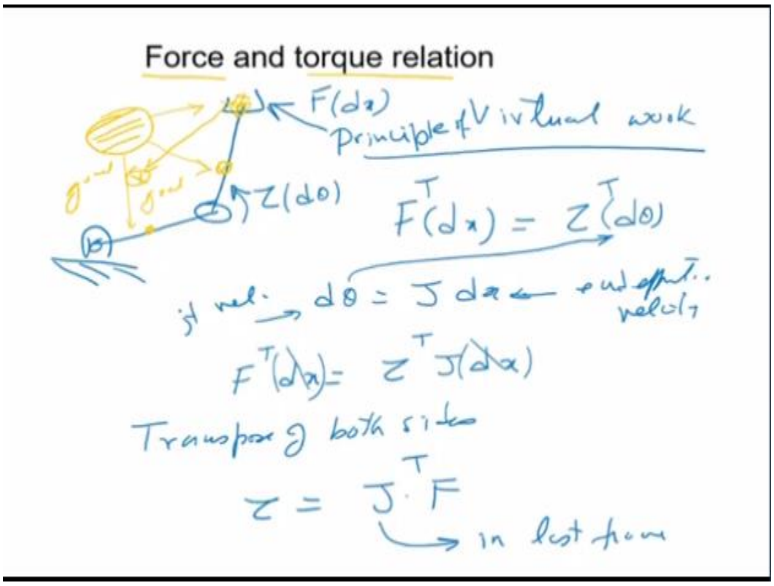
So, the path which is the closest to this so this I can take this difference and say that is the deviation from path that may call that deviation e and minimize that. So, whichever path gives me the minimum e, let us say I divide this into 20 points. And for each of those points, I actually find what is the deviation from this straight line and then I do the $\sum$ of that and say that this is my total deviation. And that is how I am going to minimize my path length.

So, you can write a program in C language or any other language math lab and actually find the values of η and ζ. So, this is sometimes we call tuning and you have to tune this because there is no other way of finding that out. Now, it can be with respect to path length, it can be with respect to energy, the moment when we go to 3D cases, we will see that in 3D when the robot starts climbing up, then there is less energy consumption goes up.

So, in 3D applications like in autonomous vehicles and unmanned vehicles, then it is a question of energy, which one takes less energy? And so, in this case it need not be that it is a straight line is the best path whereas you need the part that is going to give you the new energy. When we come to 3D applications, we look at this. So, you can have different you can have a combination path length plus energy plus distance obstacle. So, these are all cost functions which can be used.

But what I am trying to emphasize here is that there is nothing like what is the optimal value of this, what is the tighter it depends on the values I put and if I increase it for example, and make $\zeta = 100$, $\eta = 10$ what will happen? It will get pulled faster. So, every point in the retro field will be subject to a force which is much higher, so if this was my force F_1, this is my force F_2 and $F_2 > F_1$, so you move faster in terms of energy in terms of velocity.

(Refer Slide Time: 16:55)



So, far we looked at mobile robots, now in the last class; we were looking at several manipulators also. Now in the case of serial manipulators, we need to find what is the relationship between the forces acting on the end effectors? And that effect of it on the joints. So, for example, what I am trying to say here, is that we have and 2 link manipulator. So, there is a force acting here, let us say there is a force affecting their, so affects F, I observed and there is a torque and this is causing a torque. What is the relation between torque and F?

Why? Because we can convert all the forces acting at different points in the manipulator, for example, there is an obstacle here. So, let me draw an obstacle here, there is an obstacle and I want to go from this point to this point that is my goal. So, what I can do is this force which is acting on this robot is because of this, there is a force acting, so each of these points these are forces acting, so this repulsive forces are forcing the manipulator away whereas the attractive force is pulling the end effectors towards the goal point.

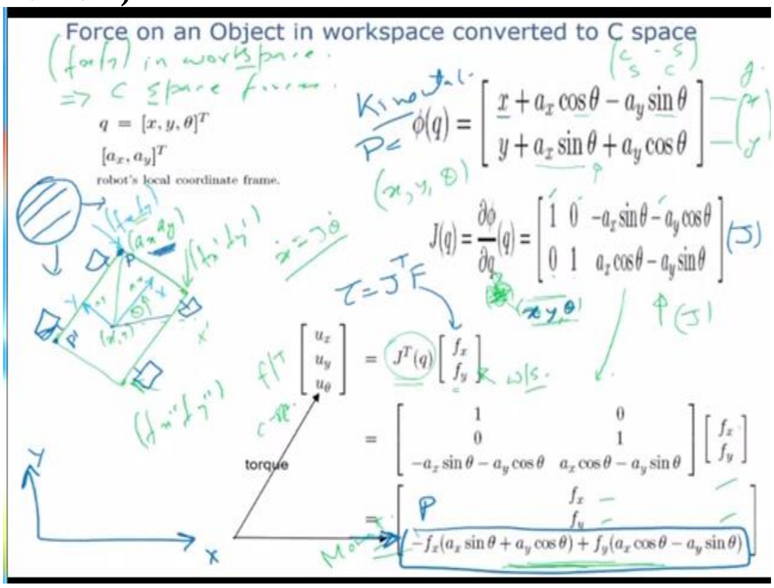
And again, it is a potential field method so I am going to assign forces potentials, take the gradient and then find the forces acting on these points, maybe point here, point here, point here and then forces find the effect of the forces on the torques and then some other torques then go to my C space. So, essentially, what we are trying to do is we are finding the relation between forces and torque acting at the joint. Now, if there is a force F which is acting at the end effector here let us say this force produces a small deflection dx.

Now, correspondingly from virtual work, this force is equivalent to a torque at the joint which has a small deflection dx joint and from virtual work principle. So, principle of virtual work, we can say that $F^T(dx) = \tau^T(d\theta)$. Now, we also know that $d\theta = Jdx$, where this is my joint velocity and this is my end effector velocity, so this we do know. So, now I can take this d θ and put it here. So, what I get from the equation $F^T(dx) = \tau^T Jdx$.

So, I have dx, dx on both sides so, I can cut this and I can cut that they cancel each other, now If I take the transpose of both sides then what we do get is $\tau = J^T F$. Now please note that this J is in last frame it is not in the first frame that means the forces acting on the last frame. So, this gives me a relationship between end effector force and the joint torques. Now, if there is end effector force acting or there are different forces acting on the different links.

Then we can convert them into the joint torques and I can sum them up and then say that these are the forces acting on my manipulator in C space. So, please note that this is done in condition space so what I mean is there is a force acting which can be effects F $_y$, F $_z$.

(Refer Slide Time: 20:54)



Now, what is the use of this? the use of this essentially we can use this principle when there are multiple forces acting an object. Let us say this is a mobile robot, it has 4 wheels and this is my global frame x axis, y axis and with respect to this, the coordinate of this point is (a_x,a_y). And the local frame is here $x'y'$. So, now what I can say is that suppose there is an obstacle here, this obstacle is applying a force f $_x$, f $_y$ at this point so let us call this point P.

Now, this obstacle could be applying a force on other points also so, this is another force let us say P'. So, it could be applying some of the force on that point P'. So, what I can do is I can take all these forces f $_x$, f $_y$ acting at point P convert to torque take the force acting at P_1 or

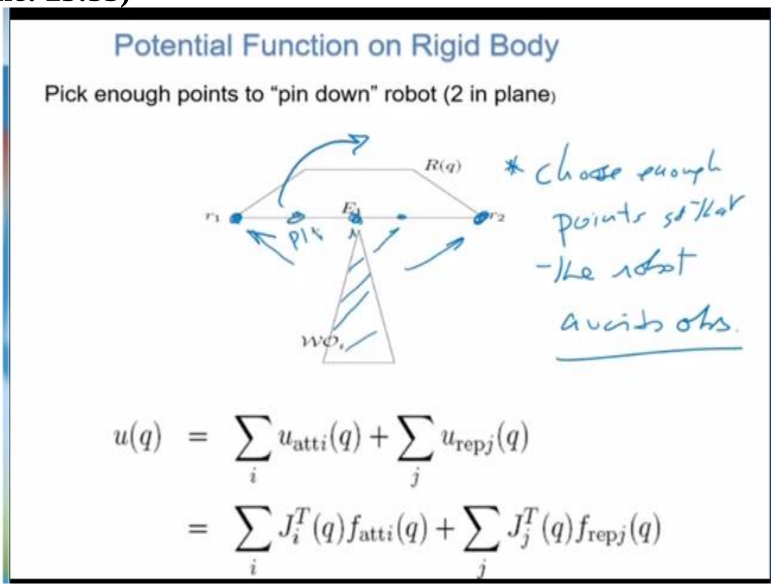
P' convert that to torque and add the moment. And once we add moment then I can say that the C space where the robot is a point all these moments are acting on the robot now.

So, how do I do that I basically find the kinematic relationship so this is my kinematic relation or geometric relation where because the these 2 are rotated, the local frame is rotated by an angle θ then x the coordinate of the point P is given by x in the x direction is $x + a_x \cos\theta - a_y \sin\theta$, where a_x and a_y are the coordinates of the point P in the local frame, then if I take the derivative of this with respect to dx, dy and θ, then what I get is this is my geometrics.

So, first I take derivate with respect to x, then I take the derivative with respect to y and then I take the derivative with respect to θ. So, what we do get is my Jacobian matrix J and then I will use my relationship which we just saw $\tau = J^T F$, this is my F and that is my τ. And what we do get is? I get the moment which is acting which is the last column here this one on the point P.

Similarly, I can find the moment which is acting on the point P_1, P', P_2, P_3 where whichever point and then I can some of them up and say that this is the force this the moment acting on the robot, which will tend to rotate the robot in taking in some of the direction because of this obstacle.

(Refer Slide Time: 23:33)

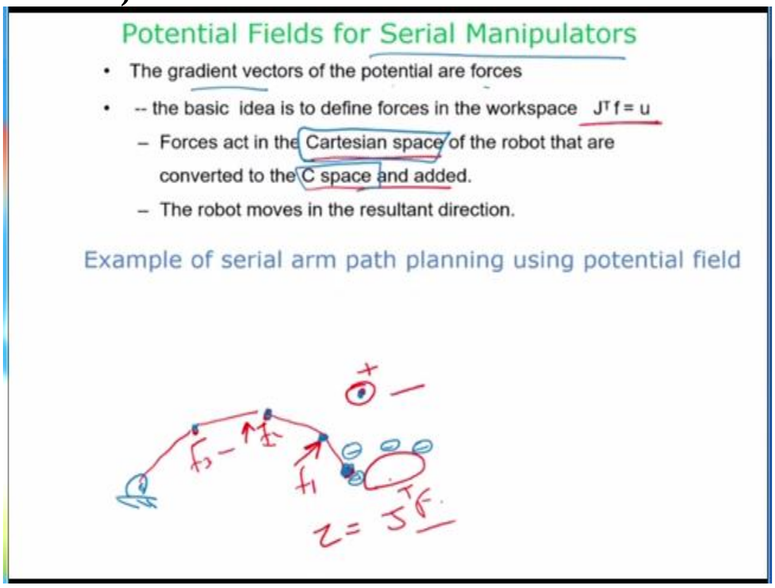


So, what we do or what we apply this for is by choosing different points onto the robot and seeing what forces are acting on those points. So, for example if you have a very large robot like this and we have a sharp obstacle. Now suppose the force acting here it is P_1 upon $P_1$and the force acting there is P_2. Then what we can do is I can convert these forces at P_1 and P_2 into a moment which is acting on the robot and move it accordingly in space.

Now, please note that suppose I say there is only P_1 and the robot will tend to rotate like this. What will happen is? This part may go and hit the obstacle now, so I have to choose enough number of points, so that the robot does not hit the obstacle, robot avoids obstacles. So, you have to choose again choosing the point is up to you, I can choose this point. And I can choose this point, this may not be enough because this might hit. So, the force is acting, their force is acting here. But what can happen is it can hit this point now.

So, it might be a good idea to choose many points here and see what are the forces acting on all of them, convert them by this equation, sum them up and then see what is the force acting on the robot and move it accordingly in C space.

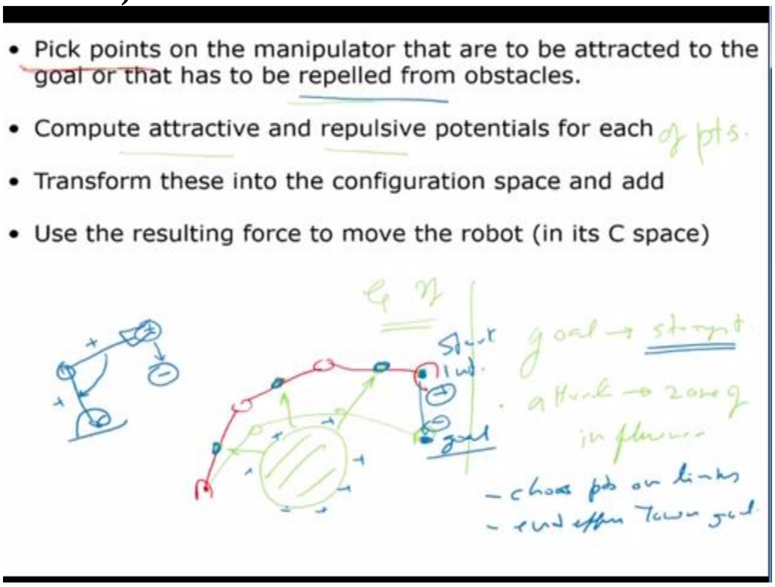
(Refer Slide Time: 25:05)



Exactly the same thing we do in the case of serial manipulators. So, in the case of serial manipulators we know that the gradient vectors of the potentials are forces. So, the basic idea is to define forces in the workspace that means if this is a manipulator I choose this points one point here, one point here, one point here, and say these are the forces acting because of an obstacle. Now, suppose I want to go from here to here, what I will do is? I will assign potentials here and let us say this is minus, this is plus, this is minus, this is minus.

So, the obstacle will tend to push the robot away and the forces will be acting at all these points that we have chosen now, you have to choose those points and the vector is one point maybe the joint centres another point and see what are the forces acting on those joints, convert them into C space forces and then go to C space and then do the planning. So, forces acting in Cartesian space of the robot are converted to C space and then added it then the robot moves in the resultant direction.

(Refer Slide Time: 26:13)

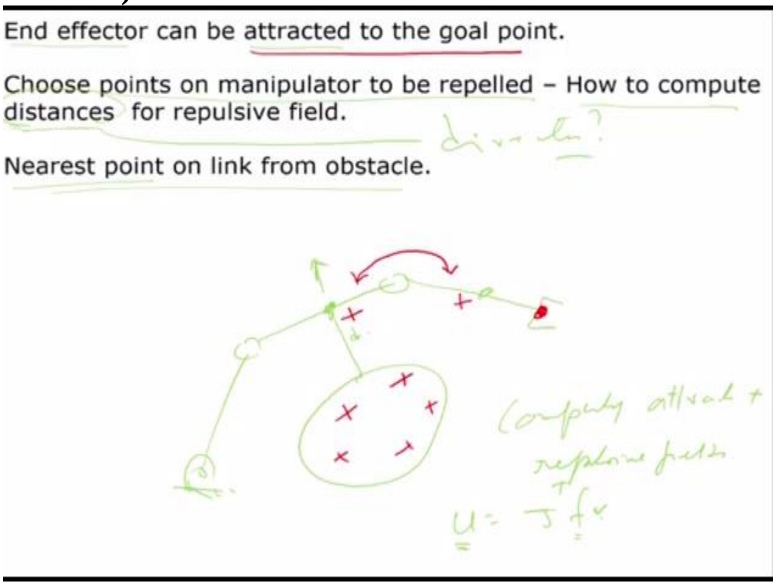


Now so it is very important here that we pick points in the manipulator that are attracted to the goal and those that are repelled from the obstacles for example in this particular case suppose my goal is here and my initial point is their start point is here then I have to choose because the end effector has to go to the goal. So, if I say this is my plus and that is my minus then this is plus, plus, plus.

Now, it is not enough to say the end effector should go to this point because in the process, these links can go hit the obstacle. So, I have to also choose points on the links now so you have to choose points on the links so goal has to be attractive. Now choose points on links to avoid hitting the obstacle, end effector towards goal. And the end effector has to be attracted towards the goal. Now, each of these links is an obstacle to itself please note that.

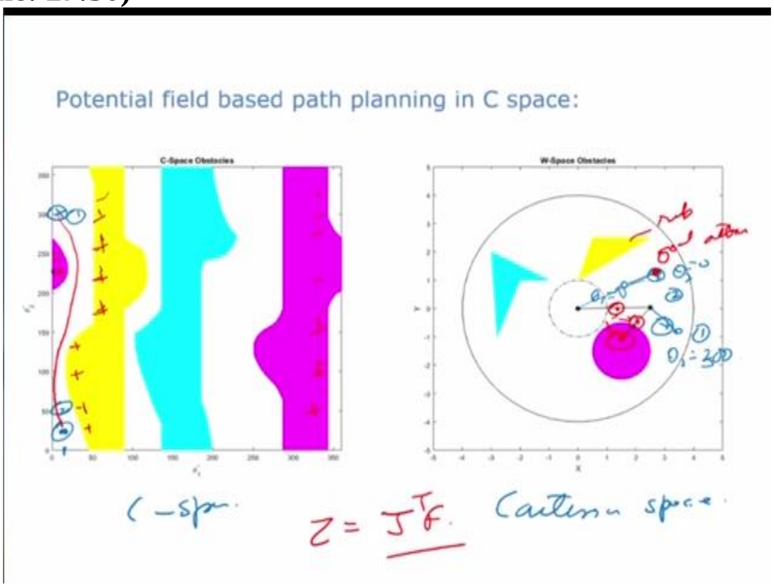
For example, if there is a robot like this and there is a link like this, and this is going it can go and hit that so a link is also an obstacle. So, if I put plus and plus they will not hit each other so they put a plus here and a goal point is minus this will get attracted here but this and this, there is an obstacle here. So, this and this will get the link 1 and link 2 will start repelling each other also, because it can happen with a link and also hit itself so one link is an obstacle for the other link.

(Refer Slide Time: 27:46)



So, this is how we basically do path planning for serial manipulators.

(Refer Slide Time: 27:50)



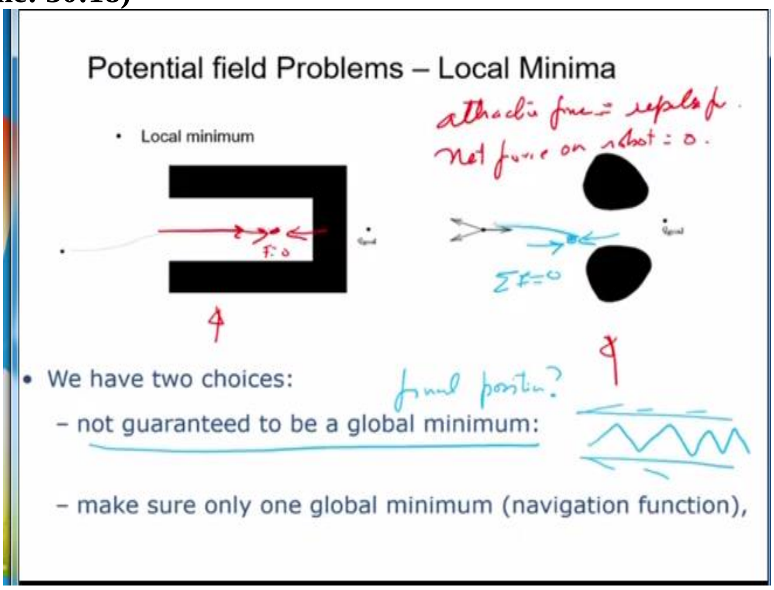
Now this is an example I had shown earlier when we talked about C space for serial manipulators. So, on the left hand side, we have the workspace or Cartesian space so this is my Cartesian space where we have a 2 link manipulator and these are obstacles. Now from here, we go to the C space. So, this my C space and we are going to do the planning between in the C space. For us say for example, in the distance between the here suppose I want the manipulator to go from this point to this point, just for example.

So, when the manipulator to go from in the bottom position it is to say $\theta_1 = 0^0$ and this somewhere here so θ_2 is about, 300^0 let us say. So, θ_1 is 0^0, and this is 300^0. So, it is somewhere up there and I wanted to go to that point where $\theta_1 = 5^0$. So, over here is attending, that is it here. And, θ_2 let us say the manipulator is like this $\theta_2 = 0^0$, so θ_2 is 0 that is 0 here so this my position 1, this my position 2.

So, from position 1 to position 2, I want to go so how do I go now I have my positions, this is the C space now. So, what we do is I can put assigned potentials at different points on the manipulator, I assign a potential onto the end effector force point there, and I assign potentials here and here. And then I assign values attractive and repulsive potentials, this is my goal point so this has to be attractive. And these are obstacles, so there has to be repulsive.

So, I see the forces acting on those points end effector and the links, use the force, use the equation $\tau = J^T F$ and then some all the torques and then come to the C space. And now when I am in the C space, then support this is my plus, plus, plus. This is the obstacle what and this is also plus, plus, this is also plus, plus, plus, plus these are plus, plus. So, the robot is going to move from here like this and come to the whole point depending on the torque which are acting in a particular direction which is moving the robot away so, this is essentially to show how planning is done in C space for serial robots.

(Refer Slide Time: 30:18)

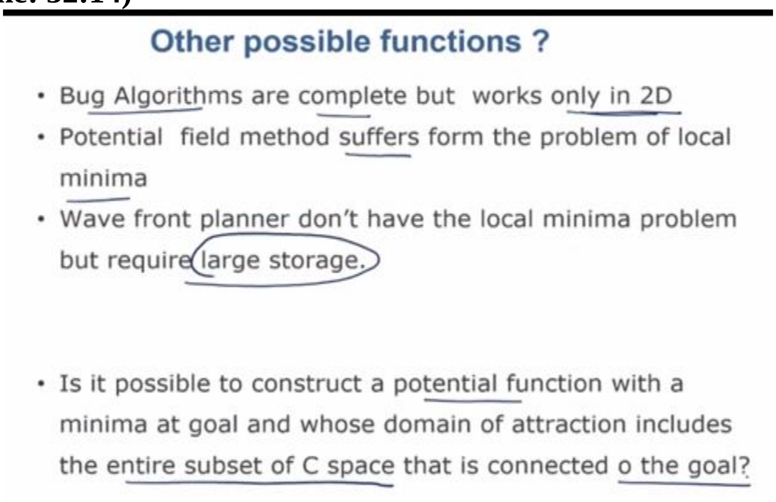


Now, we have seen that this potential fuel method suffers from some problems; one problem is this question of local minima. Now, these local minima happens when the attractive force becomes equal to the repulsive force that means so, net force active on robot is equal to 0, so net force on robot equal to 0. Now, for example, in the left hand side case we are seeing here when the robot is coming in like this and comes here. So, the force that is pulling it away and the force that is attracting it the net force becomes equals 0 there. So, the robot just stops there.

Now, it would appear that it has reached the goal because the robot has stopped, but it has not reached the goal. Similarly on the right hand side, if there are 2 obstacles placed like this then what can happen is the robot can go like this and come here to the force in this direction force in that direction cancel each other and sigma F = 0. So, the problem is that the goal the final position of the robot is not guaranteed to be a global minima.

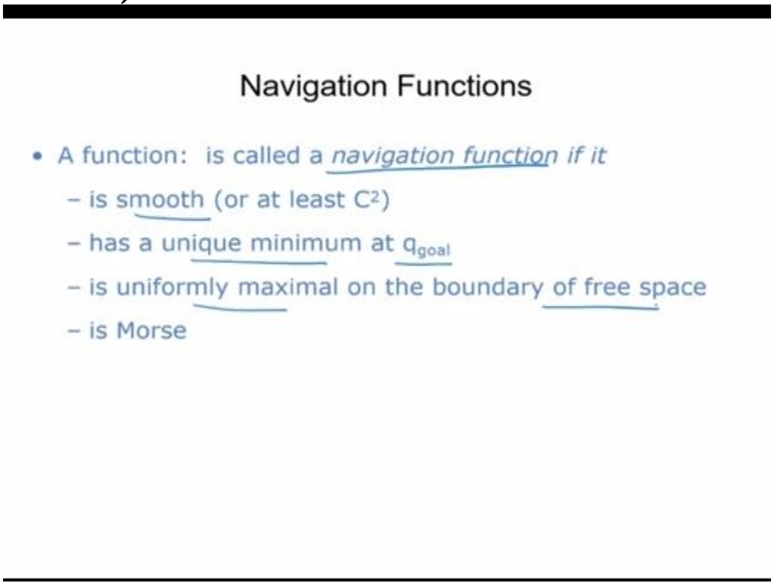
The other thing that happens very often is that if you have motion in this kind of boundaries, though it can start doing like this it can start chattering. So, this is one problem that the position of the robot is not guaranteed to be a global minima. Now, how do we ensure that the final position will be a global minima that is the question here? How can we ensure that it is not a local minima but it is a global minima. So, if you can have a function which does not have local minima or the local minima near the goal, then there is a high chance that will go to the goal only and such functions are called navigation functions.

(Refer Slide Time: 32:14)



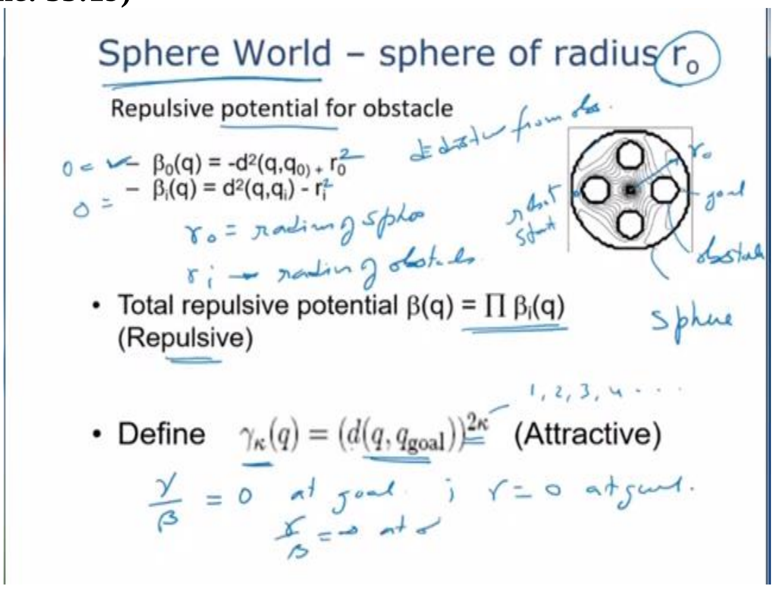
So, we have seen that the other problems that are there the earliest bug algorithms they are complete but they work only in 2D. Potential field methods suffers from problem of local minima. Wave front planners do not have the local minima problem, but require large storage so these are the problems that are there. So, is it possible to construct a potential function with the minima at goal and whose domain of attraction includes the entire subset of C space and it is connected to the goal. Now, in that case, all the local minimas would be near to the goal and if they are connected to the goal then it will go to the goal ultimately work.

(Refer Slide Time: 32:55)



So, such functions are called navigation functions. So, our function is called a navigation function if it is smooth, why does it have to be smooth because you want to take the derivative and it must be contiguous. Therefore, it has to be smooth, next it has a local minima at q goal is uniformly maximal on the boundary of free space then is Morse.

(Refer Slide Time: 33:19)



So, once that is function is this one which we call the sphere world, now in the sphere world, you can imagine this to be a sphere. So, this is a sphere that is a spherical and these are obstacles, these also spherical like ball. Now, this is my goal point and it is the robot start point. Now there are local minima is in there, so, there is a chance that what will get stuck somewhere?

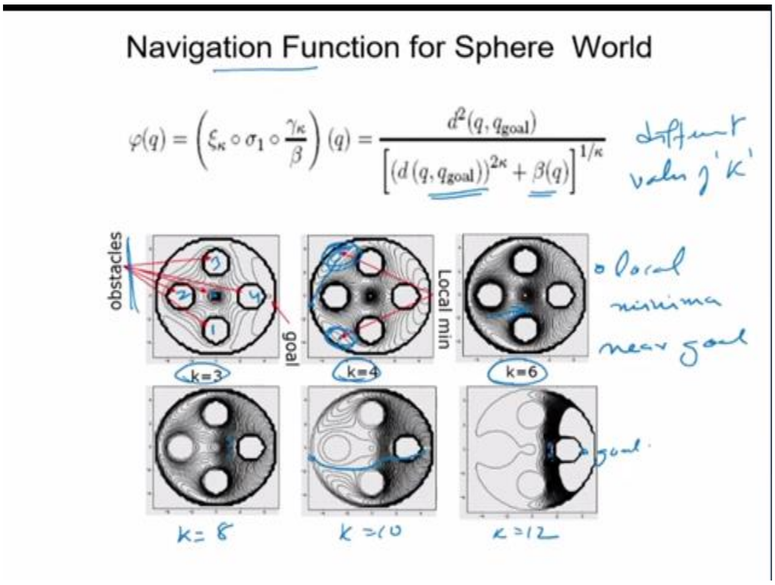
So, we have to define a function such that the local minima will tend to move towards the goal so that ultimately goes to the goal. So, how do we define such a function? So, for example, a function would again be an attractive potential and a repulsive potential like. So, the repulsive potential here is given this way, so, this my repulsive potential for the obstacles were r_0 is the radius of the sphere. So, this may r_0 then r_i is the radius obstacle so, r_1, r_2, r_3, r_4.

Now, β is the repulsive potential which is $\beta_0(q) = -d^2(q, q_0) + r_0^2$. And this is the repulsive potential for the sphere the larger sphere the outer one and for an obstacle it is $\beta_i(q) = -d^2(q, q_i) + r_i^2$, what is d? d is equal to distance from obstacle, so d is the distance from the obstacles. Now, when it comes to the boundary of the outer sphere what will happen when this one becomes 0 and it comes to the boundary of an inside obstacle, it will also again become 0 because the d^2 will become equal to r^2.

Now so now the total repulsive potential is equal to the ∑ or the sum of β_i, this is the total repulsive potential acting on the robot. Now, what is the attractive potential? The attractive potential we are defining like this $\gamma_k(q) = (d(q, q_{goal}))^{2k}$. So, depending on k we can put 1 2 3 4 this attractive force is going to change.

Now, we know that if I take the ratio of this $\frac{\gamma}{\beta}$ now, this is equal to 0 at goal, why? Because γ will become 0 at goal because the distance between the present position of the robot and the goal so, at goal is 0. Now, this goes to infinity otherwise at the surface of the obstacle so $\frac{\gamma}{\beta} = \infty$ at obstacle boundaries.

(Refer Slide Time: 36:37)



So, if I can define a function like this now we are defining the function this way our cost function maybe, which is $\varphi(q)=\left(\xi_k\sigma_1\frac{\gamma_k}{\beta}\right)(q)=\frac{d^2(q,q_{goal})}{\left[(d(q,q_{goal}))^{2k}+\beta(q)\right]^{1/k}}$. So, I am defining my navigation function for sphere field this way and then seeing 4 different values of k how does it behave, how is it behaving? So, how is the control lines moving where are the minimum of minimum or minimum constant are they moving towards the obstacle so, these are obstacles as written here.

So, these are obstacle 1 2 3 4, so it is an obstacle here also. Now, you can see that these are local minima this is the control lines or the potential function. So, these are the local minima there no, that means the robot is moving from here, it can go there and get stuck there because these are all local minima yet. Now, this is for value k = 3, k = 4, k = 6. Now, what you are seeing is that if you look at the control lines, the control lines are moving says that the local minima are getting pushed in that direction. we said that the local minima near goal.

Because if you can ensure that now k = 8, k = 10, k = 12 what we see is that? At k = 8, please note the local minima this points, so, these are moving towards the goal. So, they are all coming in this direction if you can see that there are moving in this here. So, k = 12 everything has been squeezed here. And the goal is here. So, we have bought the local minima near the goal now, which means that when the robot is moving, what is going to happen, the robot is going to move like this, it says come here so it has to only go there no.

So, the chance of getting stuck in some other local minima somewhere else is not there so, such functions are basically call navigation functions. And for each problem, you have to write your different navigation function, it is not like there is a standard navigation function. So, for different problems depending on the problem, we have to define our navigation function also. So, this was an example of our navigation function. Now, let us look at a few simulations of the potential field function. A few examples of robots which are avoiding obstacles.

(Video starts: 39:04)

So, here we have a robot which is represented by the circle, there is the start point that is my start point that is my goal point. Now, these are once you can consider this to be the top view of a room. So, these are all boundary walls and these are all walls where the robot cannot hit. So, we are going to assign attractive potential to the goal repulsive potential to the walls and the robot also has a potential such that the goal attracts it. So, the robot will be attracted towards the goal and it will go and reach the goal.

So, let us have a look here. So, you can see that the robot has started off and just go back a little bit. So, the robot is started from there. It is moving towards the goal so it goes almost in a straight line towards the goal. It comes near the wall gets reflected or deflected and then it crosses the wall and then goes to the wall. Now, we can place the robot in different positions in this one room and then. So, let us see if you are positioning the robot in different positions here. So, let me look this out.

So, the robot in this particular case the robot is positioned here, the goal is still the same point. Now simply by assigning potentials what we are doing is the robot is moving you can see from the narrow passage is going out towards the goal and reaches the goal. Now, we choose another location again, you can see the robot has moved. Now, we can choose another location which is on top that is avoiding the top wall coming and then reaching the goal. Not every point if you are not very carefully, it is getting difficult if there is an obstacle there.

So, this is the way the potential field method works by assigning different potentials to the robot and to the goal point and making it more so this is another case where the robot is there does come when it is reached the goal. Now, this is another example that we are coming to this I just showed you so let us move to the second one. Now this is an example of a mobile manipulator system so this is basically an example of a mobile manipulator system. Now this mobile manipulator system basically means that there is a mobile robot.

So, let me draw a robot here is a mobile robot has wheels and there is a manipulator on top let us say there are something of freedom and there is a mobile robot this full mobile manipulator system. So, there is a mobile robot plus a manipulator. Now, the objective here is that mobile manipulator starts off from there. Now it is there and the whole point is here at that point now, the right this is showing the top view the right side view is showing you the close up this the zoom view so you can see the robot there.

Now the manipulator on top is folded initially so that what it does basically is that the mobile robot reaches near goal okay and then the manipulator opens up to position the arm as designed. Suppose I want the manipulator arm to position like this having some α,β,γ. So, what it does is first their mobile robot avoids obstacles goes to near the goal, then the manipulator arm opens up and then aligns itself against the whatever is the desired position.

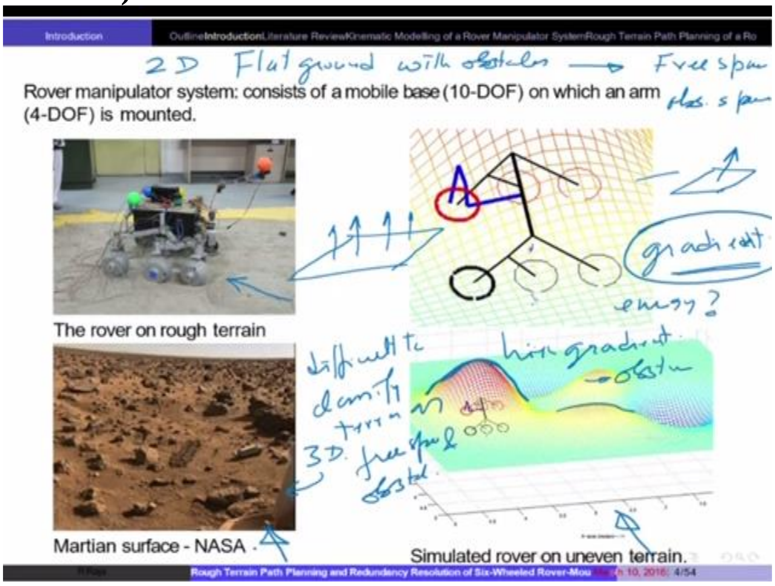
So, let us see, please note carefully the top view you can see the small dot is moving. And on the right side view what you can see is that the mobile robot is moving avoiding obstacles. So, it has come very near obstacles. As you can see on the right side figure it just avoids the obstacle there is another obstacle it avoids using this potential field method only using attractive and repulsive potentials.

Now the red dot is the goal position, so it comes not carefully. So, it is coming very near the red dot which is the goal position. So, it comes near the red dot now this is a zoomed up view. So, it has come near the goal position that is coming in the goal position and the desired manipulator configuration is this. So, show the end effector should be pointing in this direction like that that is desired.

So, now the robot has come very near the desired goal position now it opens the manipulator opens up mobile robot moves back and it goes and aligned itself to how does the lining align itself along the direction as desired, so this was an example of a mobile manipulator system where we are using a potential function on the manipulator and the robot, but initially the manipulator is fixed is not moving only when it comes near the goal position then the manipulator is enabled. So, these are interesting applications for mobile manipulator systems.

(Video Ends: 44:39)

(Refer Slide Time: 44:40)



Now let us move on now to the next part where we are talking about applications in 3D. So, whatever we were talking about till now are all applications in 2D. 2D means to demonstrate our ground with obstacles, so how would we approach the problem in 2D? We basically said that we can divide it into free space and obstacle space the robot cannot go into the obstacle space this it is free to go anywhere in the free space that is how we approach the problem.

But suppose now you have 3D case now this is 3D, this is like a lunar Martian surface or a lunar rover or maybe the roads and 3D terrain in outdoors. Now in this case, it is difficult to classify terrain as free space or an obstacle space. Why? So, for example, I have a lunar vehicle like this rover or a space rover like this. Now, it has a special mechanism which we just call the rocker volume mechanism which can enable it to climb, so this robot can go on flat surface it can also climb.

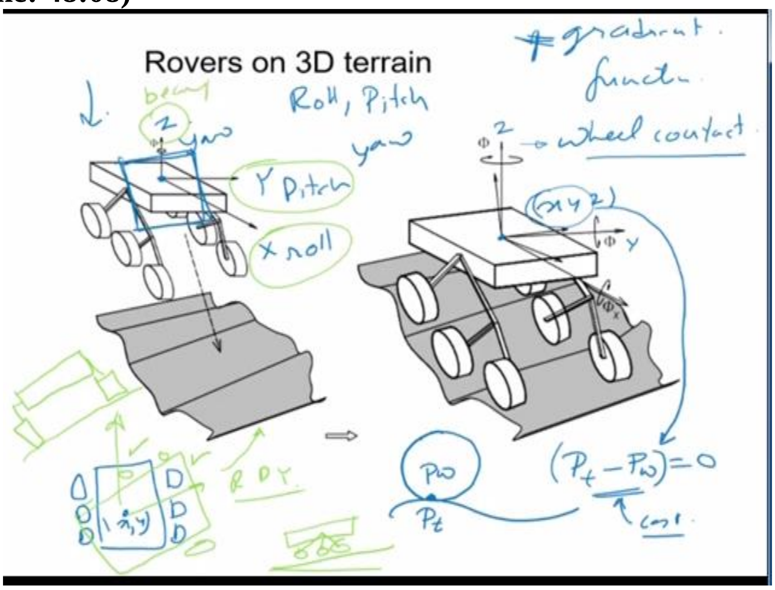
Because it has a local bogie which is this mechanism here, so this enables the robot to climb very easily. Now, because this robot can climb now when we are looking down on this surface here, this is an obstacle or this is not an obstacle. How do you decide? Because the robot can climb on top of it and go, then it is not an obstacle but it is in the 2D case the robots were not climbing obstacles there, we are just avoiding obstacles. Here we are saying the robot can not only climb but can observe it.

So, how do you define, what is free space, what is obstacle space? So, you cannot actually no you can in the sense that suppose we say that. Let us look at the right side picture. Suppose you say that this is a very steep gradient there, so I am talking about a gradient point, now there is a very steep gradient point. Now the robot obviously cannot claim that it can become unstable, it can topple over. So, that is an obstacle that is a very high gradient is an obstacle so that I can classify and say there is an obstacle.

But what about a small gradient like that, the robot can go over that. So, what we can do is? We can say very high gradient is an obstacle it should not go there but low gradients it can go there. So, basically, what we are seeing is that in 3D, we have to worry about the gradient now, which did not do in the 2D case. Now, what is the gradient if I take a grad like this, then each of these grads to will have a normal to that and the angle is normal vector max with respect to the 3 axes, it can define the grad in that way.

So, in the previous case, it was flat ground, all the vectors were like this so there was no problem. But now because the grad is not having different notations, each of the grids has a gradient point and now we have to worry about the gradient now the gradient also plays a very important part in the energy consumed because if it has to go higher, if it has a climb, it will take more energy because it is increasing potential energy mgh it is going down a slope is may take less energy.

(Refer Slide Time: 48:08)

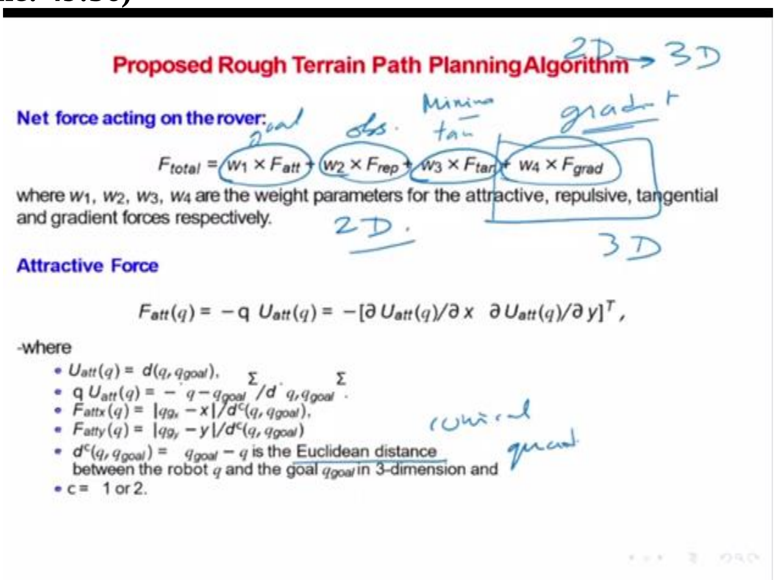


So, what we can do? We are going from 2D to 3D, we can worry about the gradient, in the potential in function, we can add the gradient also let us call it a gradient function, which will take care of the gradient it is going to, so basically I can say the robot should not climb something, it should go down because it saves energy as far as the energy is considered. The other thing is, how do we figure out wheel contact.

Now, to find the wheel contact, what we do here is there is no sensor which can give you will contact because you do not know when it is made, especially if you are doing a simulation then how do you know there is wheel contact. So, basically, what we do is? We use any kind of optimization for geometrical procedure in which we say that if this is my terrain and that is my wheel, then this contact point on the wheel, let us call it P_W and that is W terrain the P_t then $P_t - P_W = 0$ I can use this as a cost function.

So, geometrically I can find the points, so suppose this is my xyz and that is my x axis y axis on z axis. So, every point of xyz there, I can say the z is such that $P_t - P_W = 0 = 0$. Now, if P_W is below P_t that means it has gone inside the ground. And if P_W is an error, then the distance is nonzero. So, exactly when contract has been made at every x y, so this is unique for an xy. So, for every xy on the terrain, there will be a Z it where contact has been made. So, using geometrical method or by using some kind of optimization which minimizes this, you can very easily find the contact points for every xy.

(Refer Slide Time: 49:56)

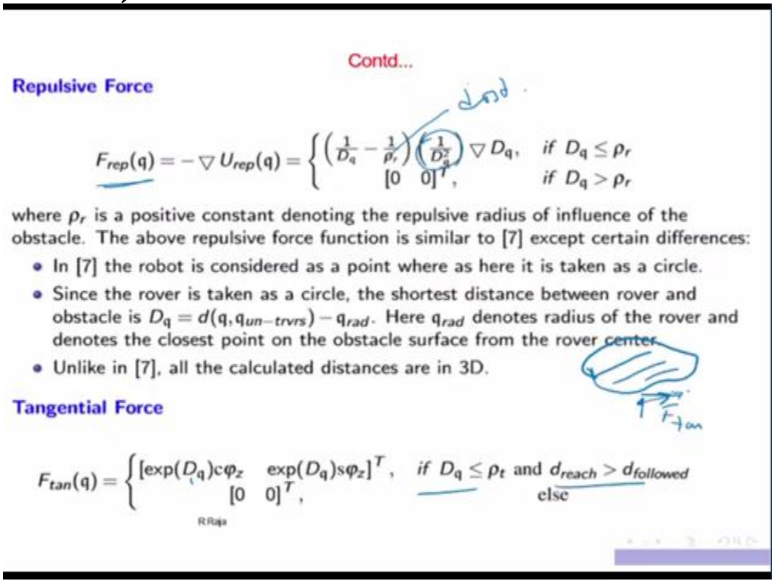


Now, in order to minimize energy what we can do is we can say that our potential function now has another component which is my gradient depends on the gradient now. So, as before we have an attractive force we have a repulsive force. So, attractive force will attract towards the goal this will repulsive from the obstacle, the gradient is the one that is proportional to the gradient of the grade we know.

So, if I minimize it will not climb try and climb it will come down now, we can also have a tangential force, so, this is tangential force basically 4 traps are minimal problem. So, if the robot gets trapped, it can you can use a tangential force which will take it away from the trap. So, this is a potential field function this is one way in which we can operate in 3D now so this is another term extension from 2D to 3D, I have this one to take it to 3D now, so this is 3D is part of 2D.

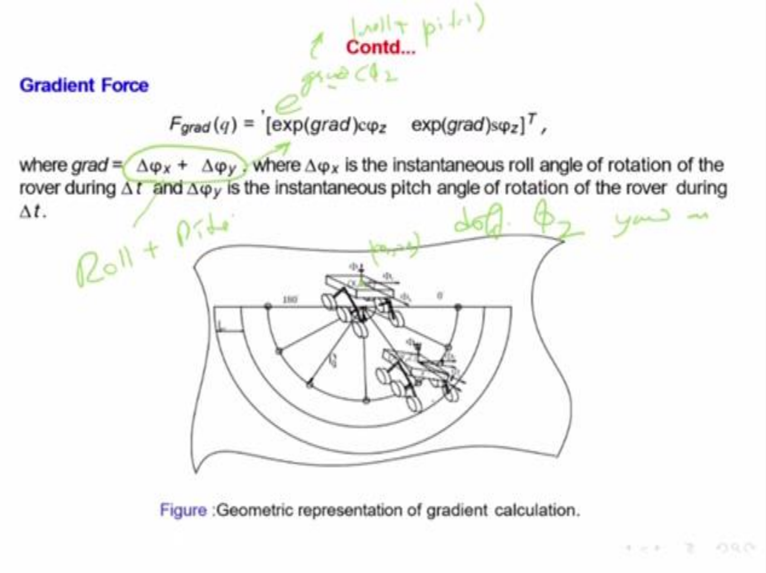
Now, I have to write my attractive force and repulsive force. Now, attractive force can be written this way so exactly similar to what we had done in the last case where d is the Euclidean distance between the goal and achieve and I can have my conical function. So, I can have a conical function or I can have a quadratic function then take this gradient.

(Refer Slide Time: 51:23)



For the repulsive I can have exactly the same. So, this is the repulsive gradient $F_{rep}(q) = -\nabla U_{rep}(q) = \{\left(\frac{1}{D_q} - \frac{1}{\rho_r}\right)\left(\frac{1}{D_q^2}\right)\nabla D_q$ so, it is exactly the same what we are using in the 2D case. Now, the tangential force can depend on case of trap. So, depending on if there is a trap whether what is not being going able to go forward but has not reached the goal then a tendency will force can be used for example, there is an obstacle like this the robot has come here it has caught. So, at this point a tangential force F 10 can be used to push the robot away and push it in this direction.

(Refer Slide Time: 52:13)



Now, what about the gradient force? Now, how we define the gradient force is let me go back to this figure is easy to explain here. So, I said that there is an x axis, y axis and the z axis. And this is also called my Roll, Pitch, Yaw axis, it is my roll axis, it is my pitch axis this way yaw axis. So, what we can do is at a particular location at this particular point, if the rope the robot or the rover is pointing in some other direction that was pointing in that direction. Then I can compute the corresponding roll, pitch yaw have between the 2 configurations.

So, if I take the top view, let me take the top view there is something like this. So, the robot is like this it has 6 wheels this makes why so, for the same xy please note the xy same what can happen is that the roll, pitch, yaw is pointing this site or it is pointing this side the roll, pitch, yaw can be different. So, it depends on the hidden angle so corresponding pointing this side or corresponding this side the xy is the same why because it is 3D terrain now.

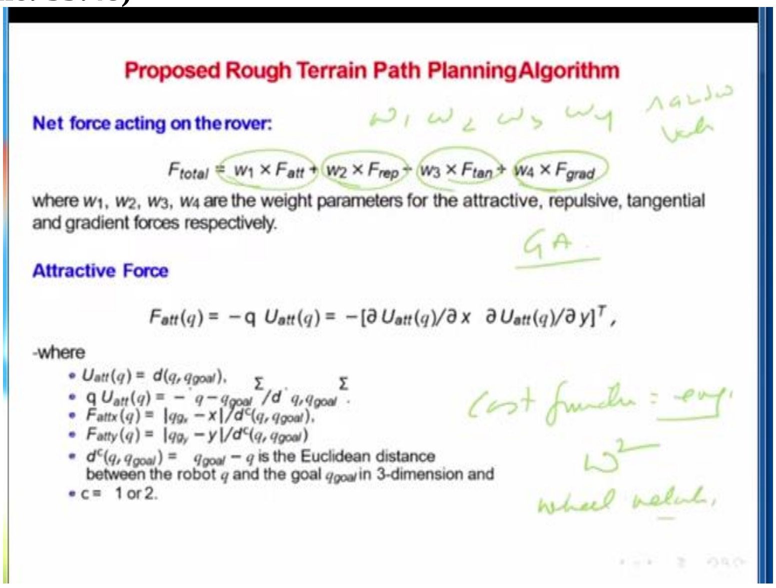
So, in different positions on xy depending on which side is facing this roll, pitch, yaw angles are going to be different. So, what is it that you want? We want that the robot should not climb too much or it should not become this is climbing one side has gone up this is one wheel let us say this is another wheel this is climbing. So, the sideways it is climbed. Now in the front direction also if it climbs this can be a front direction climb where it has 3 wheels here it is climbing like that. Again the roll, pitch, yaw angle is there.

So, what we can do is we see that this climbing and energy would depend on the roll pitch yaw, angles especially pitch and roll. So, for different yaw angles this is a different yaw, this is a different yaw which is a bearing, it will have different rolls and different pitches. So, what we are seeing here is that what we do when the robot is standing at a particular xy for different Φ z which is the yaw angle we compute what is the sum of the roll and the pitch.

So, this is the roll that is the pitch and I am the gradient force that has been used is exponential of this function. So, $e^{grad \cos\varphi_z}$ what is grad? Grad is equal to the sum of the roll in the pitch angles roll + pitch. So, when it is on flat ground what will happen? The roll and the pitch angles will be 0 because somewhere down, so it does not matter. But the moment it starts climbing they are all in the pitch angles are going up.

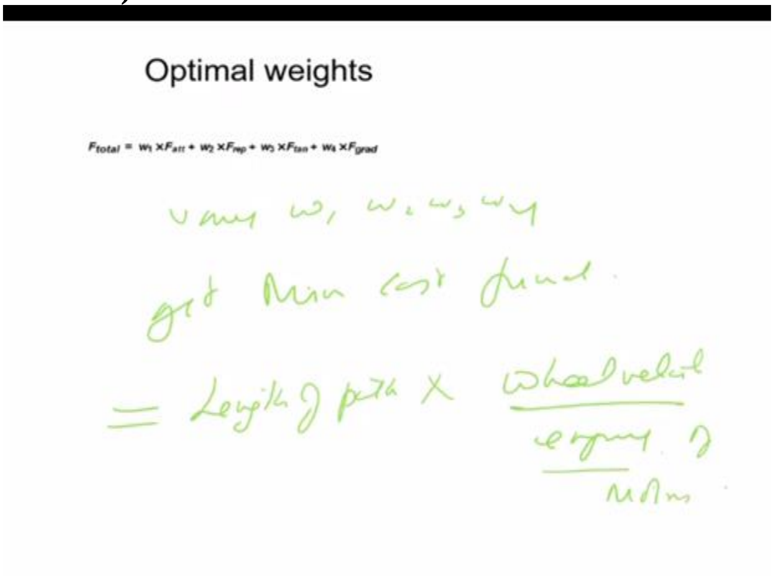
So, it is going up or it is going down, so if it is going down like this for example the robot is going down like this. So, this is my pitch angle. So, in the side direction, it was going down like that, there was my rover, what will happen is this pitch angle is lower angle. Whereas with climbing the pitch angle is higher so you can see that directly this gives us an indicator of whether it is climbing whether one side is low one side is high, but depending on that.

(Refer Slide Time: 55:48)



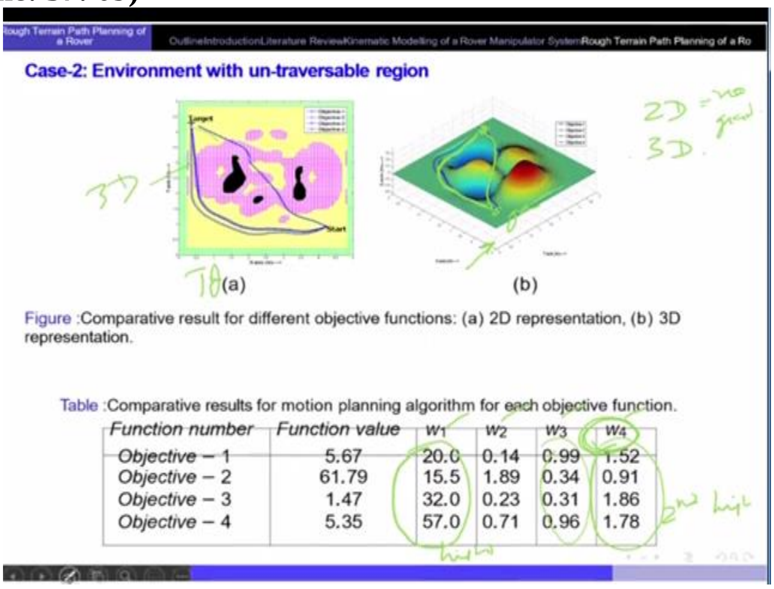
If I minimize the function then what we will get so minimizing everything with, so the attractive force, repulsive force, tangential force and grad force so, again, you see that there are constants w_1, w_2, w_3 and w_4. So, we can use different programs optimization programs, generic algorithm can be one of them where we basically have random values we can put for this w_1, w_2, w_3, w_4 and we can evaluate a cost function, the cost function can be energy, it can reveal velocities ω^2 velocities. So, we can evaluate this or we can evaluate this and say which one gives us the least value.

(Refer Slide Time: 56:34)



So, optimal weights would means vary w $_1$, w $_2$, w $_3$, w $_4$ and get minimum cost function. So, the cost function we have view that can be used is length of path into the velocities, this can be energy also and as you total energy of motors, energy of motors different of motors.

(Refer Slide Time: 57: 09)



So, if you do that, what we find is that? In this kind of terrain this the top view and this is the side view so, this is the top view and this is probably an isometric view. So, we are comparing between 2D case and 3D case. So, in 2D case, what we see in 2D case, there is no gradient, in 2D case, this might start that is my goal, robot just tends to go like this and go there. Because it is 2D case, it does not care about gradients, it does not know gradients.

But the moment you come to 3D case, what you find is that the robot this is a 3D case, the robot is avoiding all these high gradients, you can see these are very high gradients, avoiding all of that and taking a spot which is smoother, almost flat ground. So, the energy requirement has come down. And if you look at the values of the w is w_1, w_2, w_3, w_4 define that in all cases, w_1 is the maximum is the highest and w_3 was low. And this is also important, this is the second highest.

That means this shows the importance of gradient forces, in case of 3D cases that means 2D differential field cannot be applied directly to 3D because if you do not include the gradients, the robot will simply find a path which is the shortest path, but that can take very high energy we can start climbing all over. So, if you include the gradient forces, what happens is that we get very low energy and you get a very safe path.

So, today we looked at the potential field function in 3D, then we looked at navigation function in 2D then we looked at 3D. Now something to note is that whatever we have been studying so far, we found a network of paths, we found multiple paths in principle method. Then we looked at multiple network of paths for example, the sampling based methods where only diagrams we drew the visibility graph methods we did.

Now we have to answer the question of how do you choose a path length, which path are they going to take how do you decide that? So, the next thing that we will look at is basically what we call basic search, using basic search, how do you find which is the best path? So, thank you we will stop today.