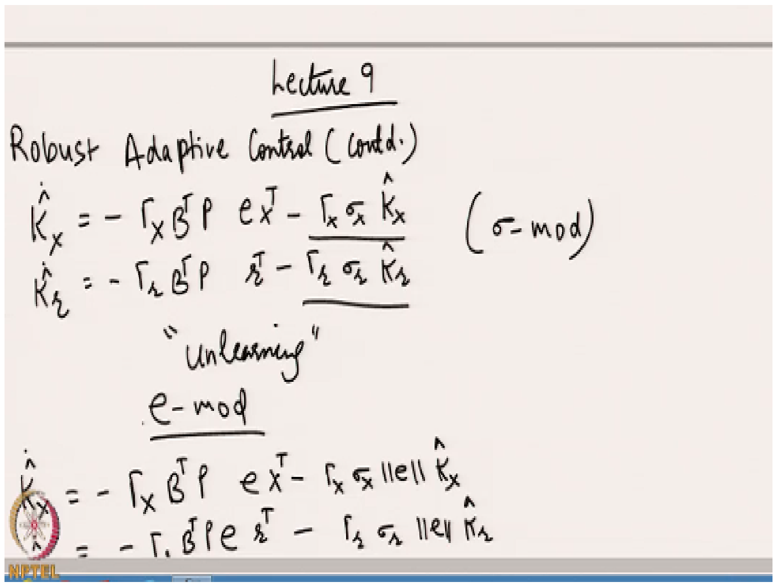
Nonlinear and Adaptive Control
Dr. Shubhendu Bhasin
Department of Electrical Engineering
Indian Institute of Technology - Delhi

Lecture – 09
Robust Model Reference Adaptive Control - Part 3

Hello, everyone, welcome to Lecture 9 of this course on non-linear and Adaptive control.

(Refer Slide Time: 00:24)

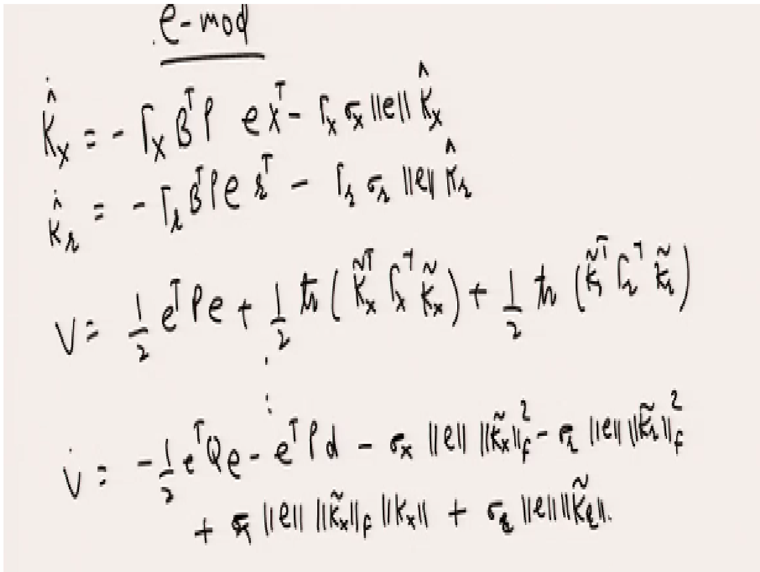


In the last class, we were discussing adaptive controllers for plans where there was a external disturbance and we saw that classical adaptive controllers are not robust to such external disturbances so that it leads to a phenomena of parameter drift where the parameter estimates are not guaranteed to be bounded which is undesirable. For any control system application because you do not want any signal in your system to be going unbounded.

And in fact we also saw that it resulted control input going inbounded which is in fact disastrous so in the last class we did 2 robust adaptive control designs 1 was dead zone modification and the other one was sigma modification where we showed that by modifying the adaptive law in a suitable way we were able to ensure that the parameter estimates they stay bounded and of course the tracking error is also bounded.

So with these two laws we were able to guarantee boundedness of all the signals including the control input. In this class we will continue with the reverse adapters control design. We will do the e mod or the e modification and we would start the projection based adapter control design so we will start off with the e modification design first. Let us go back to the sigma mod.

(Refer Slide Time: 02:24)



Design that we had done last time and if you remember this was we the modification term that we added and we call this the sigma mortification term and what this term did was that this provided the required damping which resulted in kx hat and kr hat to remain bounded so we were able to show using the Lyapunov analysis that these parameter estimates they stay bounded. So there are some limitations that we see with sigma mod term which researchers found out.

So, one of them is that when the tracking error becomes very small so for example let us look at this kx. equation so when the tracking error e becomes very small, we see that the first becomes very small and kx hat. is dictated by the second term which is a sigma mod term. So remember that sigma mod is added to ensure that the parameters stay bounded. So, it is main role is to robustify the adaptive laws.

Whereas the first term is our main learning term which is trying to learn the ideal value of kx hat. So, ideally we would want the first term dominate and the second term is just in case the parameters try to drift away make sure that the parameters stay bounded. So, in the case where the tracking error become very low we see that the second term becomes dominate. And that results in kx hat converging towards 0.

So, which is something that we refer to as an unlearning phenomena. So, unlearning is when kx hat has already learned some value some estimate for kx and at that time the tracking error becomes very low and so the kx hat is updated based on the second term which makes sure the kx hat converges towards 0 so whatever value it has learned it should basically has to unlearn that so this is of course not good.

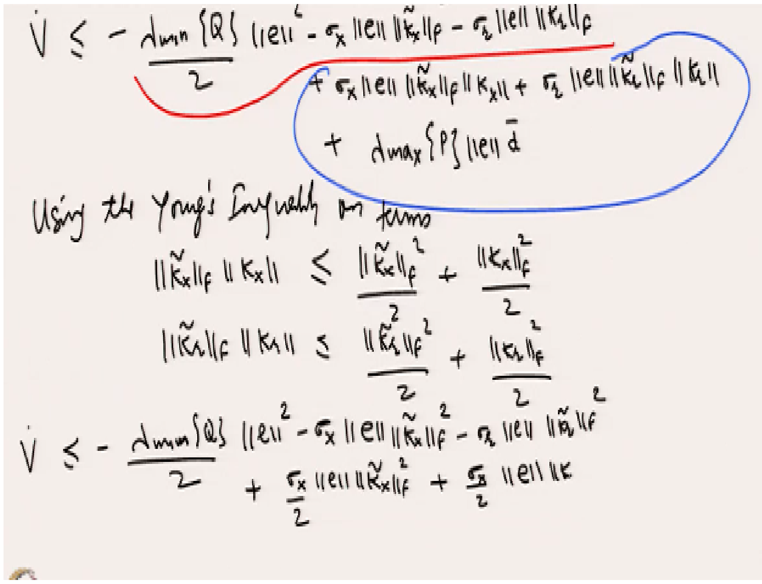
If you are looking for good performance from our adaptive laws. So, to prevent this another modification was proposed which is called as the e - modification or the e mod. So, here in addition to so we slightly modify the sigma mod term and we use the tracking error norm this and also so there was a small error. So this was the modification which first proposed so if you look at the second term sigma mod has not been modified.

And there is the norm of the tracking error which is now multiplied with the second term that is the only modification. The slight modification results in both these terms becoming small and the tracking error becomes small so intuitively you could thing that this would prevent the unlearning behavior and result in better performance. So, let us see if we can prove or stability using e- modification term.

Of course if you simulate you should be able to see some performance improvement using the e-modification term as compared to the sigma modification term because intuitively it seems that the unlearning phenomena would be curtailed. So, let us go back to the upper now stability analysis and see if we can prove stability so we use the same upper now function candidate. We follow the same steps that we had previously.

And we end up we take the time very bit and we end up with this expression - sigma r norm of the kr square+ sigma x by the way these norms are all Frobenius norms, this is the Euclid norms of the tracking error vector. We have kx2 and if you remember we had this residual term in case of sigma as well but we have tracking error multiplied with this.

(Refer Slide Time: 09:31)



This is the expression that we get once the substitute for the error dynamics and also these update laws or kx hat or kr hat of the e mod. The task here is to see if we can prove stability still so here we see that we can further upper bound v. like this and then this disturbance term also we can upper bound that by using the upper bound of the disturbance so+ lambda max which is the maximum Eigen value of the limit x p times d bar times norm of e.

And d bar is the upper bound disturbance and then we have so that is also collect similar. So, let us move this term at the end and let us have the negative terms in the beginning because it is just adds us in figuring out what terms are negative and what terms are positive so that we can dominate the positive terms with negative terms collect the negative terms together. So, these are the negative terms.

And we have the positive one which is associated with sigma x or kx and the other one which is associated with kr and of course we have the disturbance term as well. So, now we see that there is a set of negative terms which are given by this and there is a set of positive terms which are given by this. If you can somehow under some conditions of course if you can somehow dominate these positive terms with our negative terms.

Then we can show that v.<0 which is desire objective. But if not then under some conditions if we can say that these negative terms dominate the positive terms then we can try and work on those conditions then and see if those are reasonable and we are able to prove stability. So, what can we do with the positive terms. So we can use the youngs inequality again so we had discussed that last time so using the Young's inequality on terms like this one.

We say that this is < =norm of kx tilde/2+ norm of kx the tilde is not there on the kx yeah okay similarly for kr tilde using these upper bounds we can further upper bound v. as -lambda m square - sigma x . So, these terms are good terms and these terms are our friends and we have to eliminate the positive term using these negative terms somehow if we can do it. So, now we use the youngs inequality here and separate out these two terms.

So, we have sigma x norm of e kx/2+ sigma x/2 e kx square+ sigma r forward to norm of e k r sigma r of 2 and kr+ of course we have the disturbance term. So, now what we can do with this is that we can combine we can combine some terms like this this one and this one we can combine and then we can combine these 2 terms and end up with this simply. I can in fact now so if you combine these two term what we will get - sigma x square 2 of norm e times.

Norm of kx where in similarly for this term so then we can take norm of e outside and see what we get so we hope that we would be able to prove that v. is < =0 under some conditions. So, let us see what those conditions come out to be+ sigma x/2 kx tilde square+ sigma r/2kr tilde square - sigma x/2 kx square - sigma r/2 kr square and - lambda max of p times d bar. This is where we very end up with and so now what can we conclude from here.

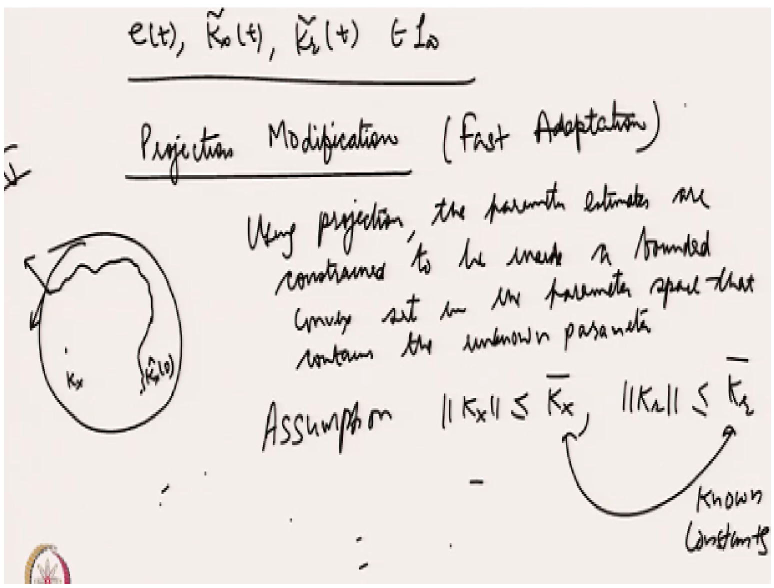
We can conclude that if it is in the brackets is positive then we can say that v. is < =0 and all signals are bounded and we are done. So, for the case that this term is positive and we do not have a problem because v. will be < =0 we can prove in single up bound but what if this is negative. So, what we can say is that let us try and put this in writing here that v. is < =0 outside the set that is called as pi.

And we defined that as e kx tilde and kr tilde such that this term in the brackets is < =0 so we can write that as lambda max of q/2 times norm of e. So let us collect the terms which are dependent on the states kx tilde+ sigma r/2 kr tilde so we have squares here right and hat if < =the constant terms which is sigma max of times d bar+ sigma x/2 of kx square+ sigma r/2 kr square.

Okay so what we see here is set is compact that is that inside the set your if the trajectories are inside the sets they are trapped the tracking error e is also bounded the parameter estimation errors are also bounded because the sum of these is <=some constant so this is some constant right. So once your trajectories are within the set with the then your tracking error parameter estimation errors are all bounded.

And outside this set v.is < =0 which means that you can say that the trajectories are again are the tracking error and parameter estimation errors are bounded so.

(Refer Slide Time: 21:55)



For all time we can say that e of t kx tilde of t and then kr tilde of t are bounded. So, this set is the compact set let us just mention that here. So, if you start outside the set then we know that v.is <=0 which means so since v is positive and v. is < =0 we can say that the that v is bounded which means that e is bounded and kx tilde is bounded and kr tilde is bounded and once the trajectories entered this compact set they stay within the set.

Because when they try to go outside v. will be < =0. All right so what we can say is that the errors are all bounded so which is what which is what we are looking for right. So, you think that e mod to ensure that all the signals are bounded and drastically or intuitively you will also see that it prevents unlearning and I will encourage that you should stimulate the adaptive control with sigma mod and adaptive controller with e mod.

And compare the performance of 2 controllers and see for yourself. If you see that e mod performance better than sigma mod especially in the region where the tracking error becomes very small because that is where the unlearning happens in the sigma mod and so this was another robust adaptive method and now we move on to the projection modification. So, both sigma mod and e mod modification terms they add as damping term to the adaptive laws.

Which result in slow convergence because they try and damp the learning rate of the adaptive estimates kx hat and kr hat and as a result of the convergence may be slow. So, for fast adaptation we would like that as much as possible we adapt based on the gradient decent update law which is given by Lyapunov analysis and only when the parameters we feel are trying to drift away that we constrain them.

So this is where the projection modification scheme is used so it is mostly going to provide you fast adaptation because It does not have the damping terms are there in sigma mod and e mod modification terms. So the basic idea is that we need to assume that our actual parameters they lie within bounded convex set. So, we need we will see how can convex set helps us here. The idea is that we want that the actual parameters say theta lies within a convex bounded region.

And so let us for a moment use kx because we are used to kx and kr so let us assume that we know a region we do not know the exact value of kx but we know that it lies in this context bounded region right and if we know that then we would like to start our updation process somewhere within the sets so we would like to start with kx hat of 0. So, time 0 we would like kx hat to be within this set.

And then kx hat of course follows the adaptation law which is unmodified law which is initially given by the Lyapunov analysis of the gradient based law so it then model tries to update based on the adaptive law and it does that continuously as long as it is within the convex set we use the same adaptive law however when it reaches the boundary that is where we feel if a condition is satisfied and if at this point which is the boundary of the of this convex set.

If the derivative kx hat. and at this point it is pointing side out of this convex set or its pointing inside the convex set. So, for example if it is trying to pointing outside the convex set we will try and limit it to stay along the boundary of the convex set. So, we apply our projection and so we project the adaptation first that it stays along the tangent. However, if at this point the derivative is pointing inside it just means that at the next instant kx hat will go inside the convex set.

And stay bounded so it is only when the estimate is on the boundary of the convex set and it is pounding away from the set that we try and project it on the tangent plain such that it either stays on the boundary of the convex set or it later moves inside. We do not want these estimates to move out of this set and that is how we make sure that these estimates always stay bounded and we also do not have any damping term that sigma mod and e mod terms have.

So, you could use fast adaptation gains high adaptation gains here you could do fast adaptation as long as the estimates are on the boundary and on the boundary pointing outside that is when they are modified. So, what we can say is that using projection the parameter estimates are constrained to lie inside a bounded compact set in the parameters space that contains the unknown parameter.

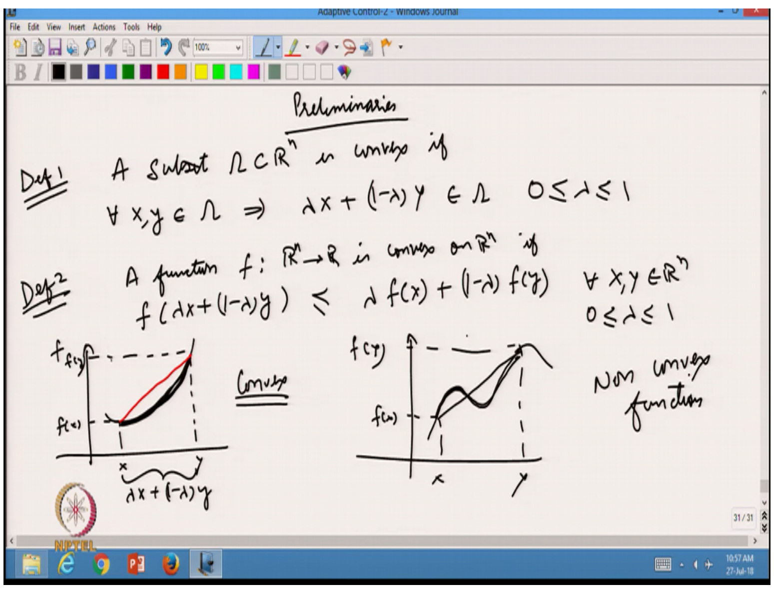
So, now if you remember we did a similar modification in the indirect adaptor control here the task was that the parameter estimates had to be projected in a region away from origin so we did something very similar we projected the estimates such that always away from the forbidden region. So, this approach is also very similar to the projection modification that we did in the indirect adaptive control case.

The only difference here is that we have to construct a convex set where the actual parameter lies so that knowledge has to be assumed that they actually parameter actually lies within this closed bounded region and then we only do it adaptation within this region. So, that is the basic idea so here they mention that the assumption we have to make say that the actual parameter if is within some bounded region.

So, kx and kr which are the actual parameters they are upper bounded by kx bar and kr bar and these two are known constant. So, this is one limitation one drawback as compared to say e mod or sigma mod where we do not assume any such assumption. In fact, in dead zone also we do not assume the upper bound on these unknown parameters there we assume that the upper bound of the disturbance is known.

So, you see that all these 4 methods that we have covered so far they have different flavors different assumptions and it depends on the application which one you want to use. So, we move on and see how this projection modification can be used to design adaptive controllers so before that we will do some preliminaries which will be useful for us.

(Refer Slide Time: 33:10)



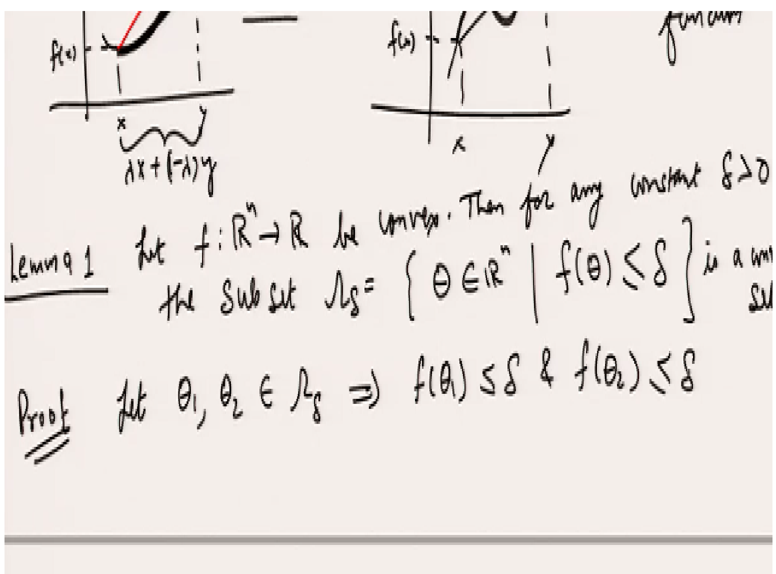
For the first preliminary is the first definition is convex set or subset omega which is subset of r and is convex if for all xy which belongs to omega and we have that lambda x+ 1 – x lambda y belongs to omega where lambda is between 0 and 1 where 0 and 1 also included so this means that if a set is convex then you take any two points in that set. All the points joining the joining the 2 points is also liable within the set so that is what a convex set is.

Definition 2 - A function f from R n and R is convex on R n if f of lambda x+ one - lambda y if there is =lambda f of x+ 1 – lambda f of y for all x y n R n again lambda is between 0 and 1 so what this means let us illustrate that using a graph of a function f. So, let us say that graph is something like this and this is the point x and this is the point y and f of x is this and this is f of y so this function is convex if all the points on the line joining x and y.

So, which is donated by lambda x and lambda y so all this region is basically denoted by lambda x+ 1 – lambda y as we change lambda we get these different points between x and y. So, if the f of lambda x+ sum lambda y which is basically this curve here the sectors segment here this is < =lambda time f of x to lambda time f of y. So, which means that now if we look at the line joining f of x and f of y which is this line for this line should always be >=this curve.

Which is true in this case so we say that this function is a convex function whereas if you look at a function like this and this is x and this is y. If we try and join the line through these three points it may be very good and let me try again so this line of course is not > =this curve here because there is a portion which is > the line so this is the non-convex function. So, we move on so let us look at lemma and this is all under preliminary stuff.

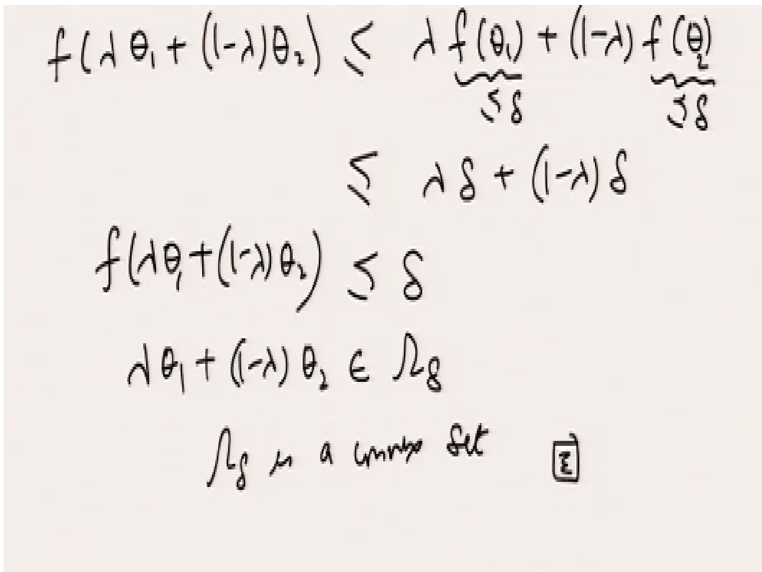
(Refer Slide Time: 38:38)



Lemma 1 so let there be a function f which is defined on Rn to R let this function be convex and we just set the definition of convex function. So, let this function be convex then for any constant delta which is positive the subset gamma which is defined as omega which is defined is called as omega delta define as theta n is to R f of theta <=delta. This is how this set omega delta is defined as so if this function f is convex then this set is a convex set.

So the proof is not really hard to follow so what we need to show is set omega delta is the convex set so which means that if we take any two points in this set and we construct a line in joining these two points and if all the points on this line also belong to this set then we say that this is indeed a convex set.

(Refer Slide Time: 40:35)

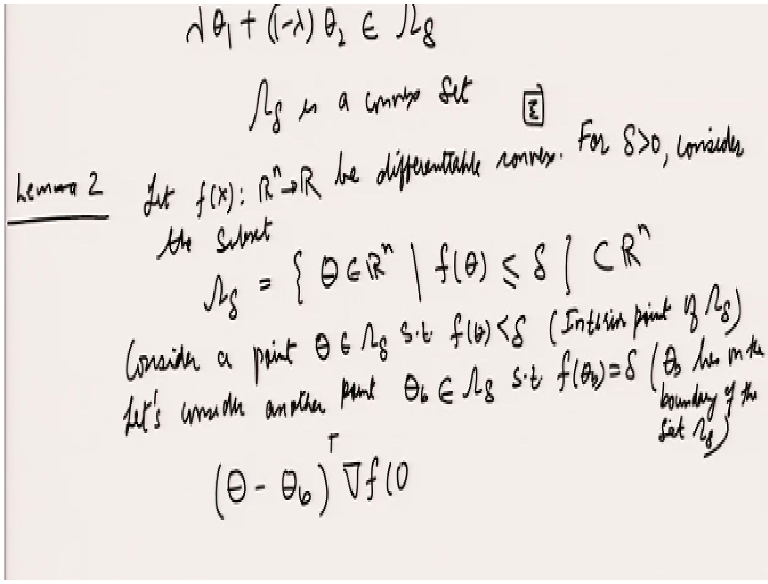


Let us consider two points theta 1 and theta 2 belong to omega delta what this means is x f of theta 1 will be < =delta f of theta will be < =delta 2. So, now we use the information which is given that this function f is convex which means that f of the lambda+ theta 1+ and one - lambda of theta 2 if there is an =lambda f of theta 1 and + or - lambda f of theta 2. We know that this is true and then we use here the fact that f of theta 1 is <=delta

Ad f of theta 2 is also < =delta because theta 1 and theta 2 are points in this set omega so then we can further upper bound this by lambda delta+ 1 - lambda delta and we simply this lambda delta and terms cancel and we also write here we have delta. So, what we are looking at is that f of lambda theta+ one - lambda theta 2 is <=delta which means that lambda theta 1+ one - lambda theta 2 should belong to this set omega delta.

Right now Theta 1 and theta 2 belong to set omega and the line joining these 2 points and all the points on that also belong to this set omega. We conclude that omega delta is a convex set.

(Refer Slide Time: 42:49)



Now we move on to Lemma 2 so in case you are wondering why we are doing this preliminary stuff and think of it as some lemmas which helps when we do our stability analysis later on to prove that the projection needs modification where in you have to construct a convex set leads to stable design. So, far we have done the definition of a convex set and convex function and we are also doing some properties related to convex function and convex sets.

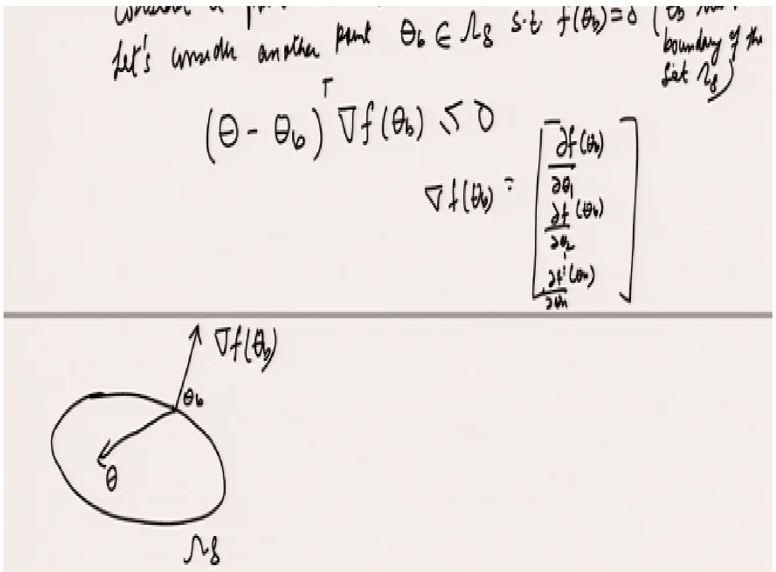
So, this will just help us when we do our Lyapunov analysis later on. So, Lemma 2 says that let there be a function f of x going from R n to R that this function be differentiable and convex foe delta> 0. We consider the subset omega delta which is constructed similarly the element belongs to R n such that f of theta is < =delta. Now let us consider a point theta within the set omega delta such that f of theta is < delta.

So, what this means is that this point theta is actually an interior point and we are saying that f of theta is strictly < delta so we exclude all the points which are on the boundary of the set. So, it is the interior point of the said omega delta. So, let us consider another point and let us denote it by theta b which belongs to the gamma delta. So, again it is a point in the set gamma delta such that f of theta b = delta so what this means is that this point theta b is going to be actually on the boundary of the set omega delta.

Theta b lies on the boundary of the set omega delta. So, what we have done is we have considered convex different symbols function f and you have constructed upon set omega delta and we have considered two points within that set one in the interiors so it cannot be on the boundary and the other point theta b is on the boundary of the set. Okay now we are making a statement to.

The lemma makes a claim which is that theta - theta b transpose gradient of f evaluated and theta b if <=0.

(Refer Slide Time: 47:40)



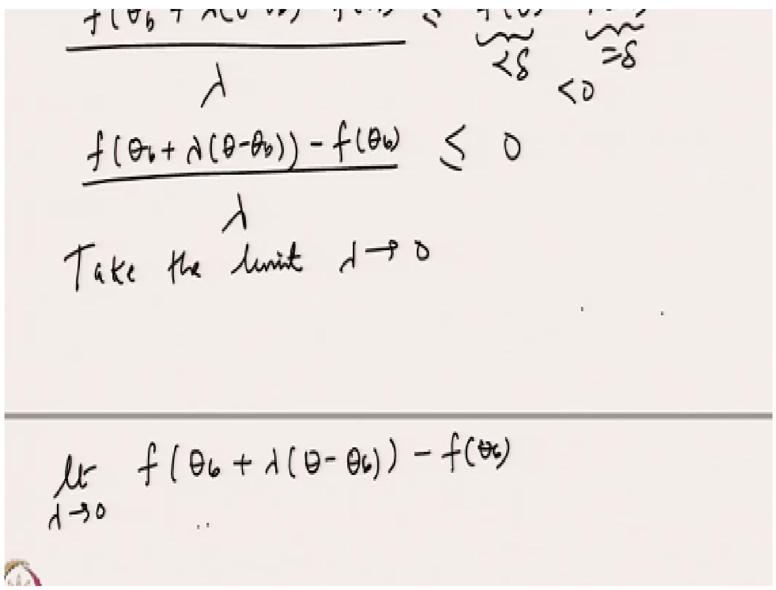
So, remember theta and theta b are elements in R n so it is a vector and that is why we use this transpose and the gradient of f is basically a partial derivative of f with respect to theta evaluated at theta b. So, this is very similar to. for. and if you look at this carefully it is basically a . for. of two vectors theta - theta b and the gradient of f evaluated at theta b and we say that. product is <=0.

So just to be clear this gradient of f at theta b is vector and column vector and del 2 and so on del f del theta n evaluated at theta b. Assuming that vector theta goes from theta 1 to theta n. So, pictorially what does this lemma say so what this means is that suppose we can consider a convex set. We consider a convex set and let us call this set as omega delta and boundary of the set. We say f of theta is =0 and on the interior we say f of theta is < =2 delta right.

So, let us consider this point theta b which is on the boundary and let us consider a point theta which is in the interior and theta - theta b will be this vector which is pointing towards theta. This is a vector that we are referring to theta - theta b so we take the drop as gradient of f. So, that the gradient of f will be the normal to this level curve so if you draw the normal to this level curve at this point theta b and it would be the gradient of f at theta b.

If you take the dot product of these two vectors we will see that it comes out to be < 0. So, this is what Lemma says that if you have a convex set and you can say 2 points one on the boundary and the other one on the interior and we construct a vector joining these two points then the dot product of this vector with the gradient of the function f evaluated at the point on the boundary will be <=0. So let us go ahead and prove this.

(Refer Slide Time: 47:40)

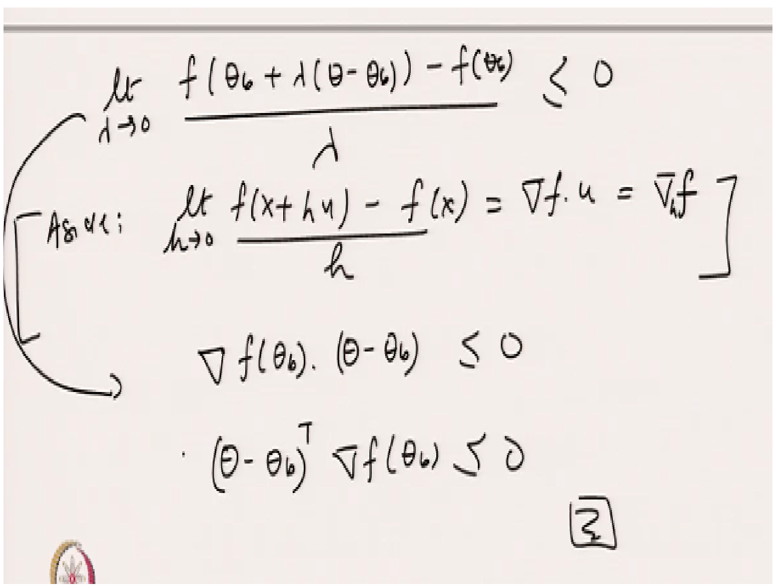


If you see it , it makes sense but we need to also prove this mathematically so since the function f is convex and f of lambda theta+1– lambda theta b is <=lambda f of theta+1- lambda f of theta b so this is definitely becomes a definition of the convex function. So, then we manipulate the system slightly and this inequality slightly theta b+ lambda of theta - theta b is < =f of theta b+ lambda f of theta - f of theta b.

So this is what we get. So, now we take this f of theta b to the other side and we divide the inequality by lambda and what we get is f of theta b+- f of theta b is <=divide this with lambda and now you get f of theta - f theta b. Since the points theta and theta b they belong to the set omega delta so we know that of course it is also given that this is < delta and this is =delta. So, so when we take the difference it will of course be < 0.

So we can say that f of theta b+lambda theta - theta b -f of theta b/lambda is <=0. Now we take the limit lambda and tends to 0 and so we take the limit increase we take the limit. So, what we get is a limit lambda tends to 0 f of theta b+ lambda theta - theta b - f of theta b/lambda is < =0.

(Refer Slide Time: 54:50)



So, what will this remain aside and this is very similar to saying limit h tends to 0 f of x+ h u - f of x. So, this is the definition of the directional derivative which is the directional derivative of the gradient of f. with u. It is also sometimes looking as gradient of u f which is same as saying that f is directional derivative of f taken in the direction of vector u. So, we basically take the and it is basically the gradient of f and we project it on to the vector u and that is what this equation is

So, we used this for the expression we got above what we can say is that this is the equivalent to gradient of the f evaluated. with-theta b. with theta - theta b and that is < =0. So, this is a. you can write this in matrix as vector notation and its basically sound as theta - theta b transpose gradient of f of theta b is <= 0 which is that what we set out to prove that this vector dotted with the gradient of the level curve who are normal to the level curve. There is always going to be =0.

So, that is where this convexity property comes into play and we will see how this lemma will help us in proving the stability in the case we have a projection modification. So in this class we have finished the e modification which is heuristically said to be better than the sigma modification term and it can prevent unlearning and we can still prove boundedness of all the signals then we move on to the projection modification based adaptive laws.

Where We want to constrain the parameter estimates to live within bounded convex region and whenever they try to move outside with project them back. So, we did some preliminary which were enable us to write these projection modification laws for the MRAC case. So, we will finish that in the next class thank you very much.