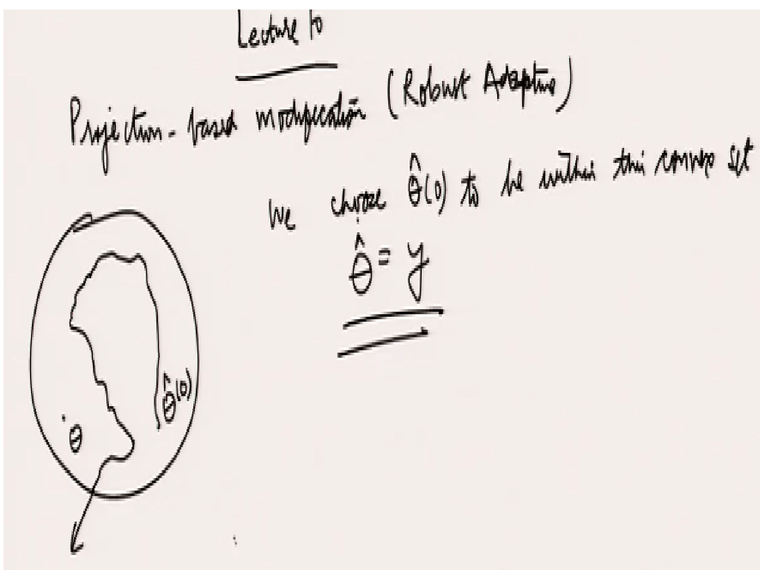
Nonlinear and Adaptive Control
Dr. Shubhendu Bhasin
Department of Electrical Engineering
Indian Institute of Technology- Delhi

Lecture – 10
Robust Model Reference Adaptive Control- Part 4

Good morning everyone welcome to the last lecture lecture 10 of this course Nonlinear and Adaptive control. In the last class we were talking about the fourth method that is the projection based modification. Because in a robust in our robust adaptive control method to constrain the parameter estimates to live within convex bounded region. So, we had done some preliminaries to start with.

So, now we will continue and see how using using those preliminaries we can actually construct projection based adaptive modification laws. And then further use them to prove that the estimates always stay bounded and they had the option of stability is preserved.

(Refer Slide Time: 01:06)



So, the basic idea of this projection based modification this is another robust adaptive term that we will use. So, the basic idea is to construct a convex set so just for convenience I am constructing a circle here considering this to be a 2 dimensional space but it could be an n dimensional space and you could construct a convex set and what we have assumed here is that this convex set is known.

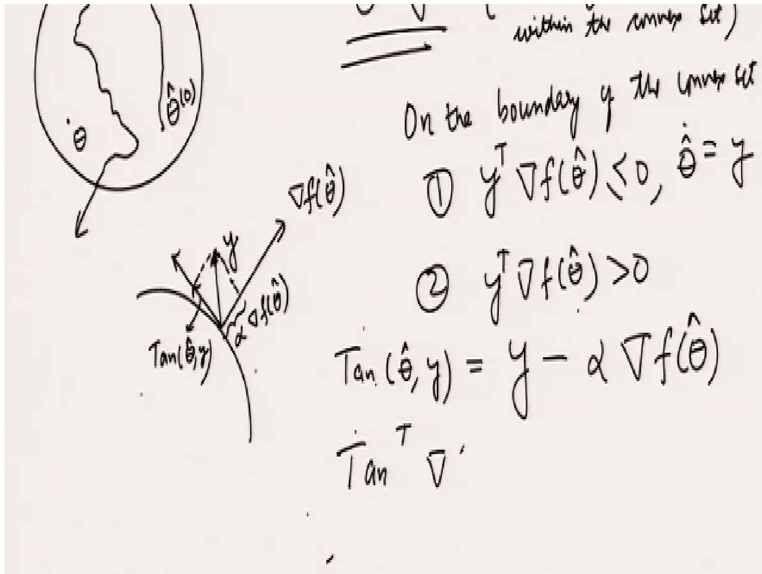
Which means that we have certain information about the unknown parameter say theta so we know that the unknown parameters theta live within this convex set. And we are trying to estimate this theta so we design update laws for theta and we denote them by theta hat and because we know this convex region in which this unknown parameter lies we want to start our updation process from within this convex set.

And so we choose at initial time say t=0 we chose theta hat of 0 to be within this convex set in which the unknown parameter theta lies. Okay so then as they mentioned in the previous class theta hat will update based on the adaptation of law derive using the Lyapunov analysis for the unmodified case. So, let us say that that law was theta hat. = say y . So, this y in MRAC case was if I remember correctly it was gamma b transpose pex transpose.

So, this was the adaptive law for kx hat right so suppose that is denoted by by y okay and so theta hat is getting updated based on this this update law and it is moving inside this convex set in a continuous way and then as long as it is within the convex set there was no problem we continue to use theta hat. =y but when it tries to come at the boundary when it did that the boundary.

And it is trying to point outside that is where we want to project it back into this convex set. Okay so how do we detect this theta hat it is on the boundary and is trying to go outside. So, for that what we do is we look at the let us look at the look at the more clearly.

(Refer Slide Time: 04:55)



So, let us say this is the boundary of the convex set and the theta hat is trying to move out. So, y denotes the derivative of theta hat at this point on the boundary. So, let us say this is at the boundary and we calculate the derivative and based on what theta hat. law that we had prescribed we find that it is pointing outside of this convex set. Okay so then just to mathematically determine that it is pointing outside we construct a gradient vector at this point.

And we take the dot product so we take the dot product of we project this vector on to this gradient vector and we see if the dot product is positive or negative. And if the dot product comes out to be positive it means that the vector y is pointing outside of the convex set. And if the dot product of y and this gradient of f comes out to be <=0 then it is either pointing along the tangent or it is trying to point it is pointing inside.

Which means that at the next instant it will either be on the boundary of the convex set or it will move inside. So, yeah so we start we start inside the convex set and as long as this is long we continue to use this a bit law till the estimate theta hat of t is within the convex set. So, we do not do any modification to it okay and then then on the boundary of the convex set there are 2 cases.

One is when the dot product which is in matrix vector notation we right it as y transpose del f theta hat is <=0 which means the direction of y is either tangential to this curve or it is pointing inside of this of this closed curve. Then we we we still do not modify this update law so we use theta hat. =y right we continue to use this update law because we are sure that in the next instant it will not go out.

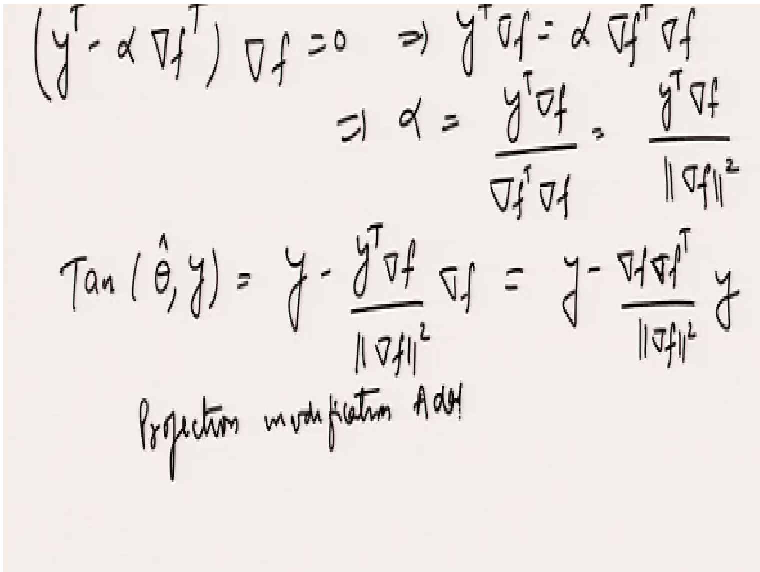
So, the second case is when y transpose del f of theta hat is >0. So, this is the case where we know that the estimate is trying to move outside of this closed convex region and that is when we want to project it back. So, how do we project it back we we do not stop the updation rather we take the projection of this y along the tangential plane. So, we construct a tangent at this point okay and then we take the projection of y on this tangent vector.

And this vector that we get is what we call the sat the let us let us call this as tangent at theta hat, y it is a function of theta hat and the derivative at theta hat derivative y. So, how do we find this tangential vector so we know that this this this vector is basically given by you can use the the law of of vector addition here and it is specifically this vector y-this vector which will give you the tangential vector.

So, this is some multiple some factor of the gradient so let us call that factor alpha time gradient of f of theta hat okay. So, we can say that this this this change in vector is y-alpha gradient of f of theta hat. So, now of course we need to know what this alpha is so we can use the fact that this tangential vector and the gradient vector are orthogonal to each other that is if you take the dot product of this tangential vector with this gradient vector will get 0.

And then we substitute for this expression for tangential vector okay and alpha is some constant okay.

(Refer Slide Time: 11:25)

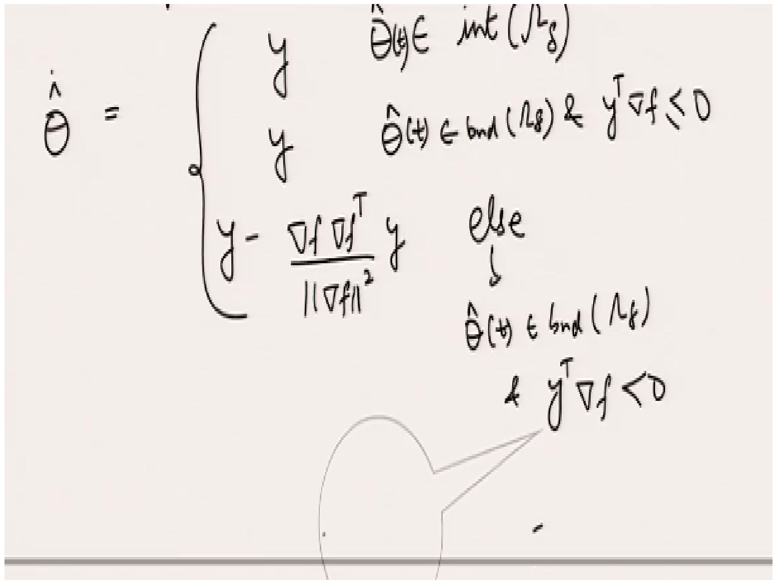


So, then what we get is y transpose-alpha then del f transpose=0 which will give us y transpose f is alpha transpose f which would give us alpha as y transpose del f/del f transpose del f which is same as y transpose del f/del f square. Okay we can say that the tangential vector at the point theta hat at the boundary is given by y-alpha y which is sorry alpha gradient del f which is given by y transpose f/gradient of f square time gradient of f.

So, this can also be written as y-radiant so since this is a scalar we can always see the transpose of this and we get and this is also so y transpose del f is in fact we can take the transpose of this and what we get is del f del f transpose y right okay so this can be moved to the other side and this entire time in the scalar and then we take the transpose of that. Okay so what have you achieved here.

So, we have actually the projection modification and adaptive law can now be written as.

(Refer Slide Time: 14:11)



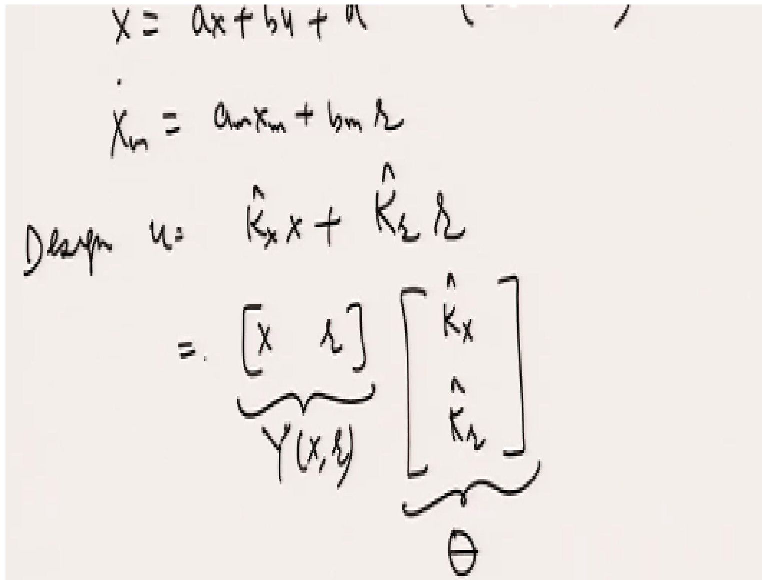
Theta hat. =y and theta hat is in the interior. So, right now I am not precisely mathematically writing down the expression for the interior of the set. I am just using int to denote interior so when theta hat of t is in the interior so how do we make sure that it is in the interior we check for f of theta hat and if it is <0 we know that theta hat is in the interior of the of the convex set. Of course first we have to construct a convex set.

But let us assume that we have constructed this set omega delta and this estimate theta hat within this convex set and then we are fine we just use the unmodified adaptive law. Then we look at the case where theta hat of t is at the boundary of this convex set. And the dot product of y and del f <=0. So, in that case y transpose gradient of f is <=0 which means that as I mentioned y is either pointing tangential to the curve or it is pointing inside.

So, in that case again we continue to use the sane update law else we modify the update law continue to use the same we modified the update law and use the projection of y on the tangential plane which is given by this vector term which we just found out. So, the expression for that is y-del f del f transpose/ del f square times y. So, we have actually taken out vectors subtracted a vector which is a multiple of y from y.

And we get this modification which is which takes the component of phi along the tangential direction and that is this the position modification that we have to do. So, what is this else condition this is the condition where theta hat of T is on the boundary of gamma delta and y transpose del f <0right transpose okay.

(Refer Slide Time: 17:15)

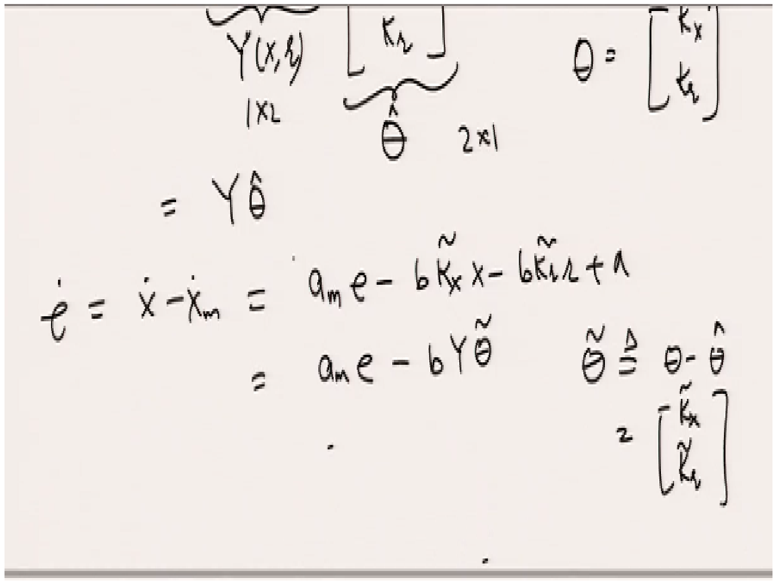


So, how do we now use that for the model reference adaptive control case. Let us look at that so projection based MRAC is what we will do now. We want to know you see how this projection can be used in the model reference adaptive control where we have some external disturbance so again let us, let us consider for simplicity a scalar case x.=ax+bu and we have a disturbance d. So, it is a bounded disturbance.

And it is a we want to follow the reference am xm+bm r are the standard matching conditions are satisfied. So, although we consider a scalar case here because we want to showcase the prediction based modification and not complicate the analysis by considering a vector case. But you can trivially trivially extend this to the to the case where you have a and b matrices okay. So, for a direct MRAC we design our u kx hat x+kr hat r okay.

Now what I am going to do with a little different from what we have done so far I am actually going to combine these two estimates into one. So, I will consider that we we have a and r then i will combine into one okay and I call this as the regressor y which is a function of x and r and I call this as theta which is the vector of unknown parameters right. So, we this is given by y theta so is 1*2 and theta is 2*1.

(Refer Slide Time: 19:57)

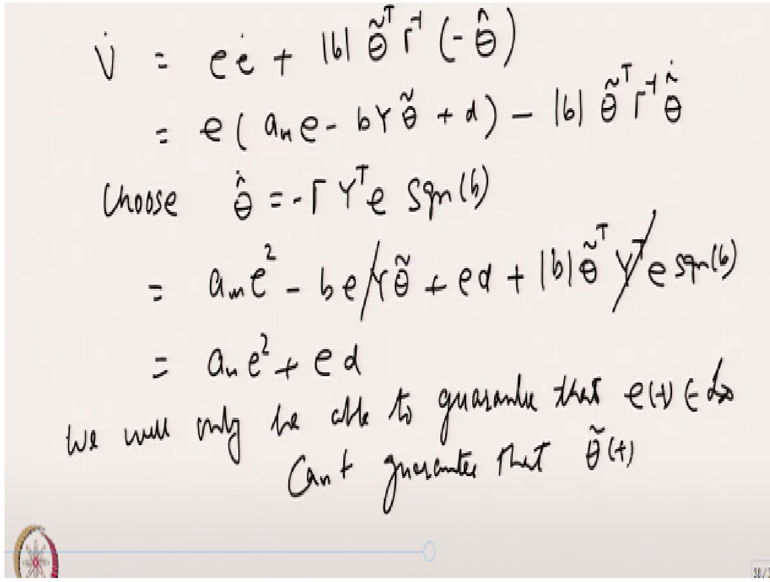


Okay u is a scalar and again y is a term which is known quantities and theta is the unknown unknown parameter vector which we need to estimate right. In fact, this will be theta hat so let us make the theta hat because an estimate of the actual parameter theta which is given by kx and kr which is the ideal value of kx and kr which found out from the matching conditions if you know the unknown parameters a and b.

Okay so again e. is given by-x.-m. and we follow a similar procedure as before and we end up with ame-d so what did we what did we get previously we got something like this plus a disturbance jump but not since we have combined kx hat and kr hat into this victor theta hat we can we can do this little differently and what we get is-b theta tilde in fact y alright okay where theta tilde is defined as theta-theta hat okay.

So, you can think of it as kx tilde and kr tilde okay + d which is the disturbance right.

(Refer Slide Time: 22:09)



So, again proceeding like we have done so many times in this course we choose Lyapunov function candidate to be some positive definite really and unbounded and a crescent function. So, for this case we conveniently choose it to be the sum of squares of the errors or so one thing I just want to mention is before we do the stability analysis is that that we can we can easily show that the parameters here.

Parameter estimates theta hat will always stay within this bounded convex set as long as initially we start off inside the set. So, using this modification you can in fact go ahead and show that mathematically also you can actually show that once if you start inside you will never be able to leave this set because of the modifications that that we are doing okay. So, our problem with parameter drift is solved if we follow these mortifications.

So, it really it works as far as removing the problem of parameter drift in presence of disturbances is concerned. The only thing that show now is that this modification does not effect the stability analysis in an adverse way. Okay okay so here gamma inverse is of course positive definite and we assume that the sin of b is known okay then we take the derivative of this now the classical unmodified adaptive law for this case would be designed.

Let us just go ahead and do the error system substitution yeah we forgot the yeah this is fine-theta tilde transpose gamma inverse theta hat. Right so if we choose theta hat. To be gamma-

gamma y transpose e signum of b then what we get is-b e y tilde+ed- and signum of b and mod of b combined to give b and then we can take the transpose of this term because it is a scalar and then this cancel with this and you are a m e square+ed.

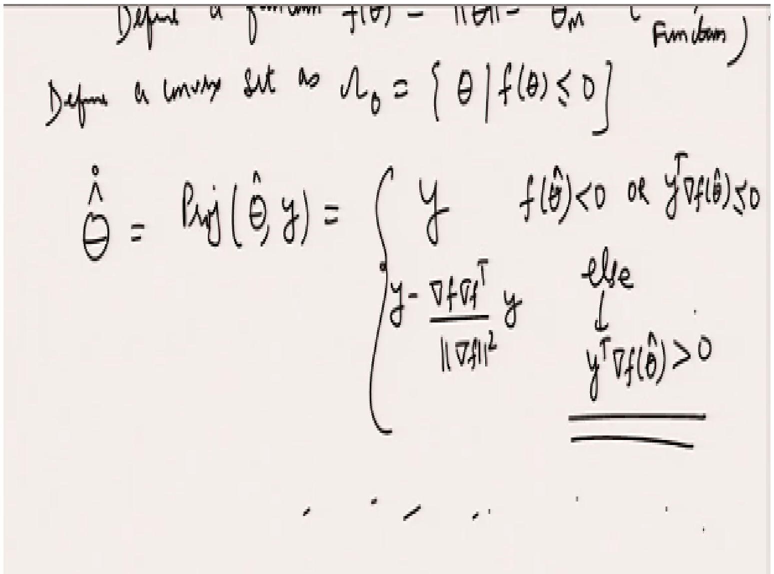
And here if you remember what we did was we were able to show that if it was very similar to the dead zone case where we were able to show that if the tracking error is larger than some threshold then v. will be <=0 which means that the tracking error and the parameter estimation errors are bounced. But once the tracking error becomes smaller than the threshold and that is where were only able to show that tracking error stays bounded.

But the parameter estimation error can drift away and go unbounded and that is where the problem started and we started modifying our update laws and added reverse elements to it. Okay so here again we will have the same problem we will only be able to show here we will if you if you follow through with the analysis like we have done before we will only be able to guarantee that e of t is bounded.

But that is not enough because we want to make sure that all signals in our system are bounded and the control is also bounded. So, we cannot guarantee that theta tilde of t is bounded or kx tilde kr tilde is bounded and so we just mention that using projection modification on our adaptive laws we can in ensure that the the parameter estimates they stay bounded. But we need to show that such a modification does not affect Lyapunov analysis.

So, remember the Lyapunov analysis you know we will have to change now because we were not using this theta hat. Update law we will not be using this at all times right it depends on where our derivative vector is. So, if it is pointing outside the convex set then of course we cannot we will be using a modified version of this. Okay so let us now modify this for the MBRAC we have already done the basics required to construct our projection based law.

(Refer Slide Time: 28:47)



So, let us modify the update law using projection so first of course we construct a convex set that is the first requirement for this kind of a projection where theta lies. So, the actual parameter lies inside this convex set. So, for that as I mentioned we have to assume that theta is<=theta m where theta m is a known constant. So, that is very important that this is a restrictive assumption in some ways.

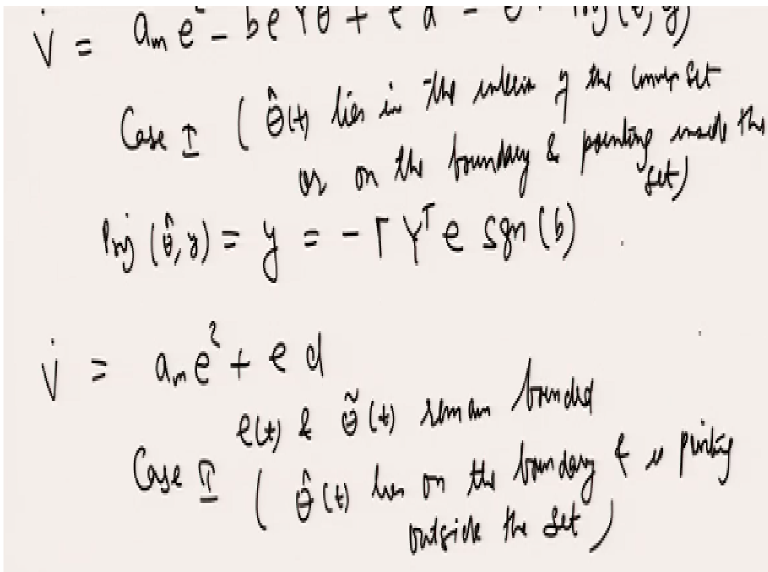
Because we are saying that we know the upper bound on the adaptive parameters which may not be available in many cases. Okay, so so how does this modification affect the stability analysis that is what we want to see okay so then we construct we define a function f of theta as theta square- so we basically square both sides and this is the convex function. Then we can using this convex function we can define a convex set like before.

So, here let us just assume let us just use delta which is delta=0 okay so this is the convex set where delta=0 okay and we use the modified update law now. So, theta hat. = so we represent it as prod so this is the projection operator which projects theta hat to be within this convex set. So, we define we write it as a theta hat, y which is given as so now we have the different cases so we have theta hat. =y which is derived from the Lyapunov analysis.

So, this is the case where the estimate lies within the convex set which is f of theta hat <0 or if it lies on the boundary then the dot product of y and the gradient evaluated at theta hat is<=0. So, if this condition is true then we use the unmodified adaptive law in the case that this condition is not true we have the else case where we modified this law like this. Okay so for this is the case for y transpose gradient of del f theta hat>0.

Which means that the derivative vector y is actually pointing outside of the convex set in that case you want to project in back okay. So, of course here reconsidered that initially at t=0 we start within the convex set so theta hat 0 has to be< theta m and we assume that theta m is known.

(Refer Slide Time: 33:37)



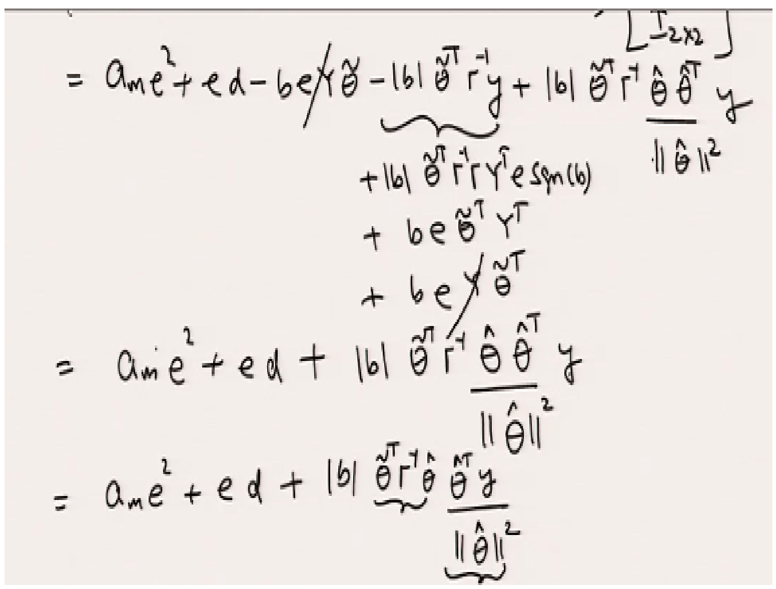
Okay so how does this modification affect the stability analysis so let us go back to the Lyapunov function that we had considered and I will come to the expression for v. which was given by am e square-be y theta tilde+e d and now we have+ theta tilde transpose gamma inverse and we use proj of theta hat, y yeah so because so we have a negative sign here okay. So, in case one so let us just denote these as case 1 and case 2 .

So, this is case 1 and this is case 2 okay so for case 1 which is when theta hat of t lies in the interiors of the convex set or on the boundary and pointing inside and this side so this is case one so in this case we know that projection of the projection operator will return the unmodified law so proj of theta, y=y which is given by what we had derived last time-gamma y transpose e signum of b.

So, if we use this as the update law we will cancel this term in the Lyapunov derivative and we will be left with v.=am e square+ed. So, this is the same situation that we had in the unmodified case the difference here is that here we are making sure that theta hat is within the convex set. So, in this case we can say that both the tracking error and the parameter estimation error they remain bounded.

So, now let us talk about the more interesting case which is case 2 that is the case where theta hat of t it lies on the boundary of the set and is pointing outside the set. So, the derivative is pointing outside the convex set so that is the forbidden region that they want to prevent here we want that it always stays within this convex set.

(Refer Slide Time: 37:50)



We will have v. as ame square+ed-bey theta tilde-so there we will use the modified update law so this is i-theta hat theta hat transpose normal theta hat square times y. Okay so I is, in fact an identity matrix of dimension I2x2 okay. So, let us simplify this equation so we will have ame square+ed-bey tilde and let us look at the first case and we will get a-tilde transpose-gamma inverse y+tilde transpose gamma inverse theta hat theta hat transpose.

Okay so if you look at just this term here let us try and simplify this so we will substitute for y what we get is+ and this becomes be theta tilde transpose y transpose and then since this is a scalar you can take transpose and nothing will change you get by theta tilde transpose okay now this term gets cancelled with this term and we will be left with a and e square+ed and then we will have this term to deal with.

Okay so let us manipulate this slightly so of course the first two terms are familiar we had obtained these terms in the unmodified case as well. We need to somehow make sure that this third term which comes as a result of the projection modification that is either negative or 0. So, we would like that this term should be <=0 so that the stability is preserved. So, okay so now we break it down into into into 2 terms this term and this term.

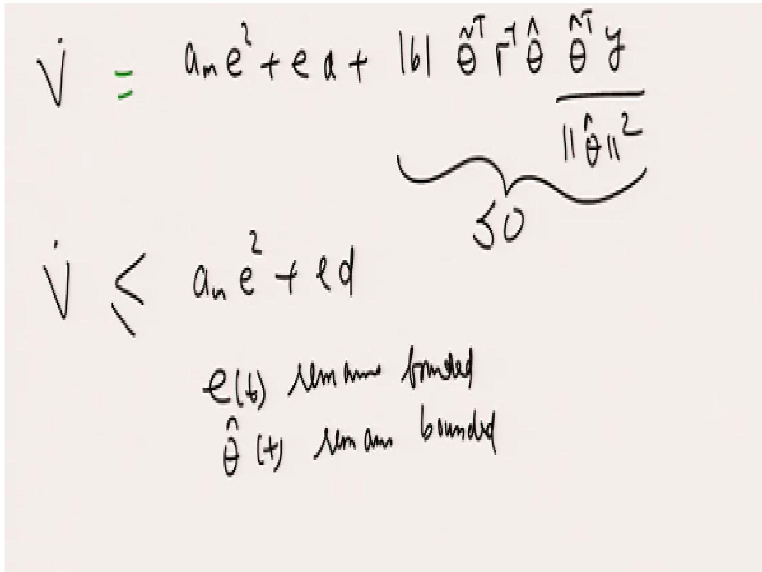
Now if we look at the second term which is this term so this is case 2 so in case 2 if we go back we see that for case 2 we had considered that y transpose gradient of f of theta hat is >0. So, what the function f that we had considered in this case is given by theta transpose theta-theta m so we take the gradient of f with respect to theta and what we get is 2 theta. So, this case 2 means that we are looking at y transpose theta being >0.

Okay so we use this information here so here this case is y transpose gradient of f of theta hat being >0 which translates to y transpose theta hat being >0. So, this term in fact >0 okay now let us look at the term the second term which is which is this term. So, i want to take you back to the lemma 2 that we had done in the previous in the previous class where we had proved that those who recall lemma 2.

Where we had proved that theta-theta b transpose the gradient of f evaluated at theta b with<=0. So, in this case theta b represents the theta at the boundary of the set and case 2 in fact represents that the estimate line on the boundary. So, we can for case 2 we can write this as theta-theta hat transpose and the gradient at theta hat I have already calculated to be twice of theta hat. So, we can just write this as theta hat<=0.

So, which means that were looking at tilde transpose theta hat<=0 so can we use that information somehow here yes we can because gamma inverse is a positive definite matrix and since tilde transpose theta hat is <=0 so this term is in fact <=0 since this is a positive term so since this term is we have proved to be positive and this term if n=0 is a combined term is <=0.

(Refer Slide Time: 45:18)



So, we can say is that let me rewrite this so this combined term is <=0 and this is v. so so we can further upper bound v. by throwing away the name the term which is <=0 so v. then becomes <=am e square+ed so we go back to the case that we had obtained for the unmodified situation and in fact here v. is more negative than the previous case. So, the same conclusions hold that we are able to guarantee.

And with so we are using this the of course we can guarantee that e of t remains bounded and because we are using the projection which makes sure that the estimate lies within the convex set for all time we also proved that theta hat of t remains bounded. So, in conclusion of this projection based modification what we can say is that projection ensures that the parameter estimates stays within bounded convex set.

And also that the stability is preserved even when a projection modification is used. So, we can say that using these 4 robust adaptive controllers starting with dead zone and then with sigma mod in emod and projection we have been able to robustify the existing adaptive controllers so that they can work in presence of external disturbances and we can ensure boundedness of all the signals in the system which is extremely desirable for any control system.

So, that takes us to the end of this 10 week course so in this course of course we have we have covered linear time invariant systems. But the idea was to expose you to the integrities of the adaptive control design and the same ideas can be extended to nonlinear systems as well. So, i hope that you enjoyed this course and you were able to learn how to design adaptive controllers using Lyapunov based methods which are not only analysis techniques but also a design tool. Thank you so much.