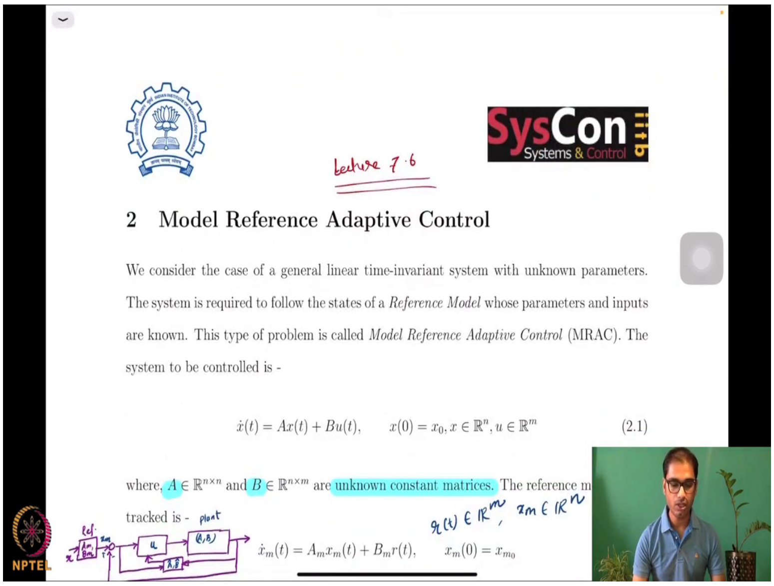
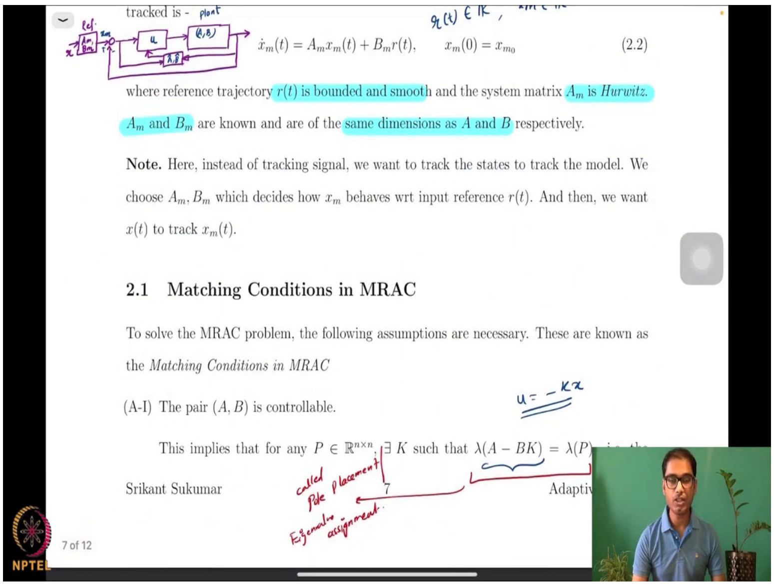
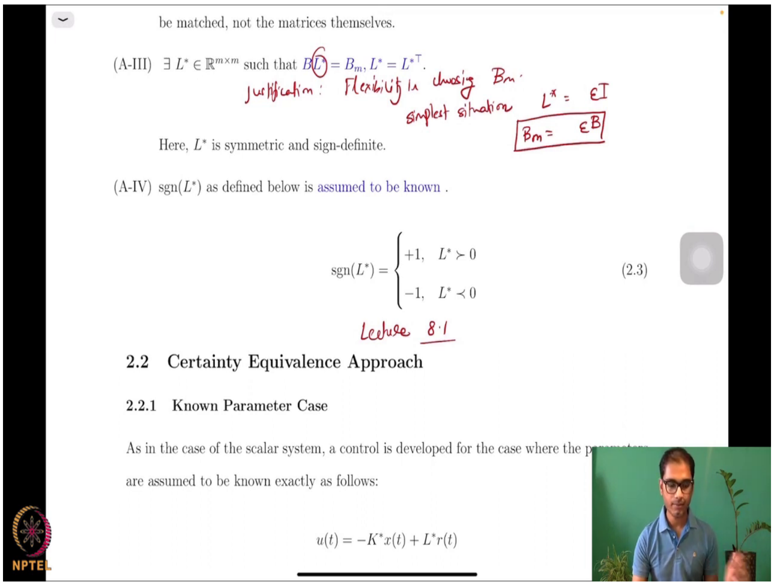
Nonlinear Adaptive Control
Professor Srikant Sukumar
Systems and Control
Indian Institute of Technology, Bombay
Week 8
Lecture No: 44
Model Reference Adaptive Control: Lyapunov Stability Analysis

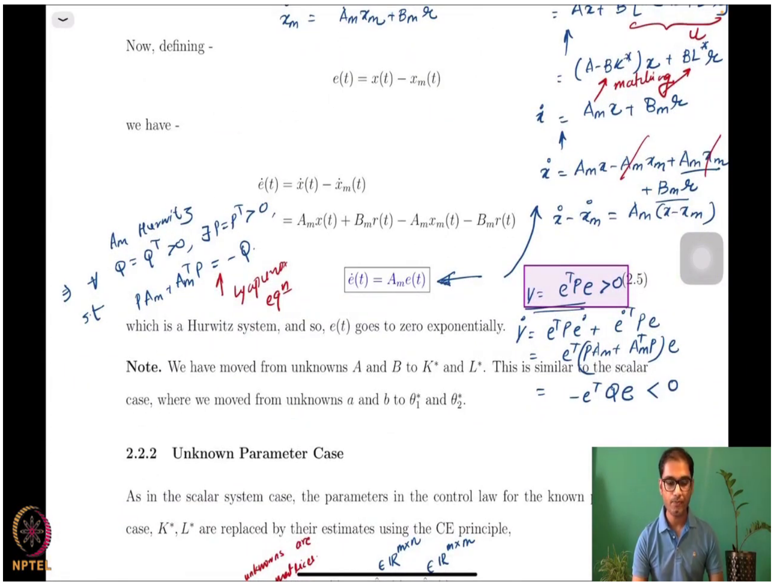
Hello everyone, welcome to yet another session of our NPTEL on Non-Linear and Adaptive Control, I am Srikant Sukumar from Systems and Control, IIT, Bombay. We are into the 8th week of our lectures on Non-Linear Adaptive Control. And we are already well underway into designing algorithms that can drive systems such as the spacecraft orbiting the earth, which we see in the background.

(Refer Slide Time: 0:55)







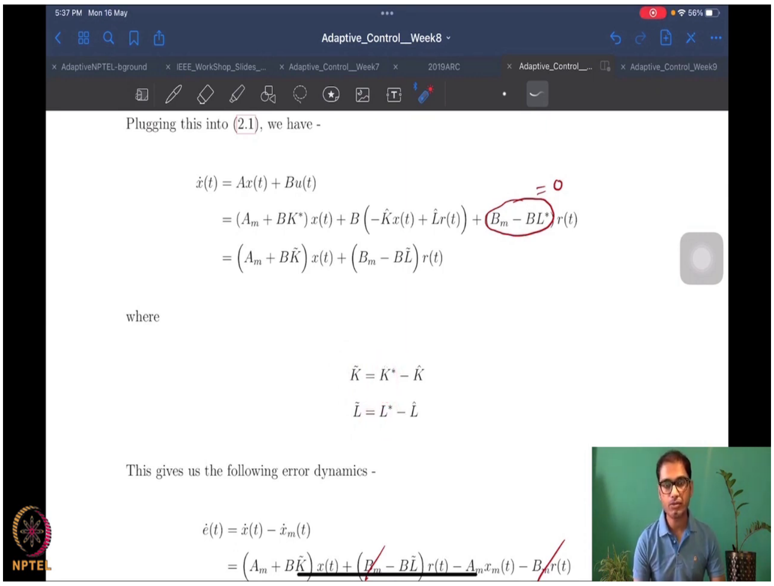


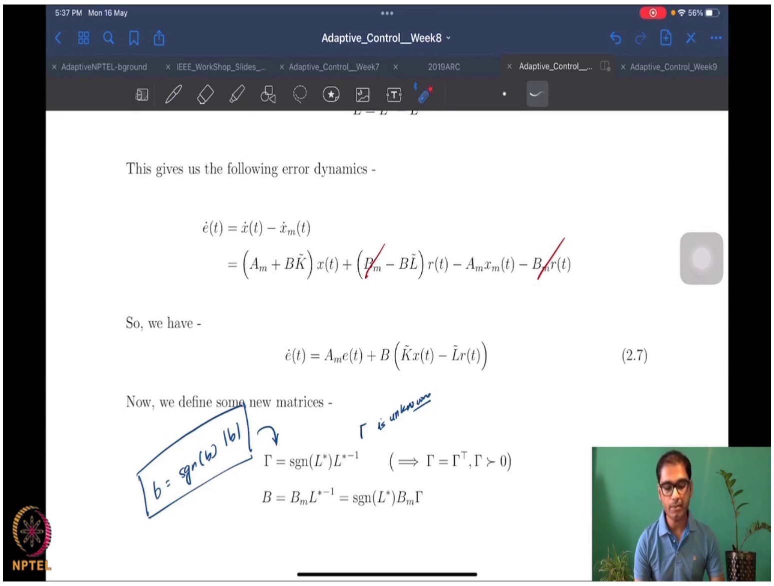
So, last time we started to look at model reference adaptive control. We already introduced the problem, in the previous week along with the assumptions and of course the matching conditions which are very very critical. Subsequently we started the design for the adaptive controller in the usual approach of starting with the known parameter case, and then of course moving on to the unknown case via the certainty equivalence approach.

The difference of course is how we chose the target system. So, the target system was in this case chosen in a very very smart way and was inspired from the reference model itself. So, this was what helped us to design the controller for the known case and of course going from the known to the unknown case was it is always very easy because we just simply apply the certainty equivalence principle.

So, that is really the trick. We apply the certainty equivalence principle in order to go from the known to the unknown case.

(Refer Slide Time: 2:16)

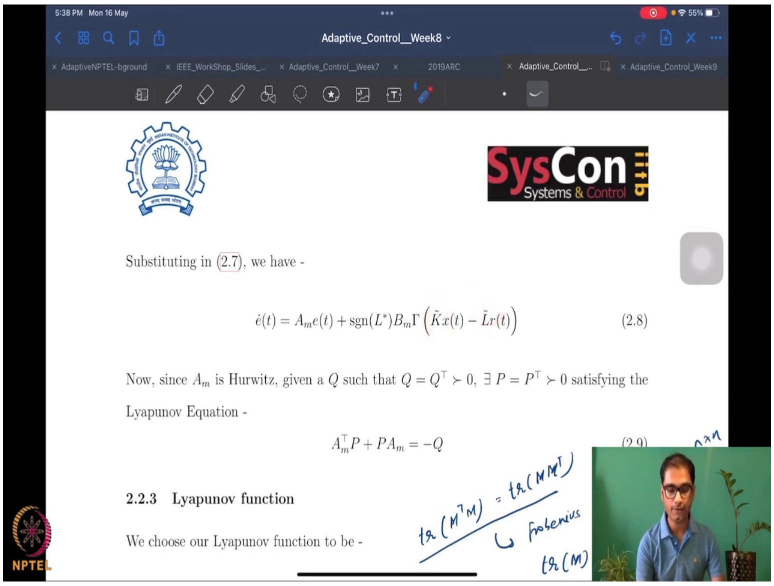


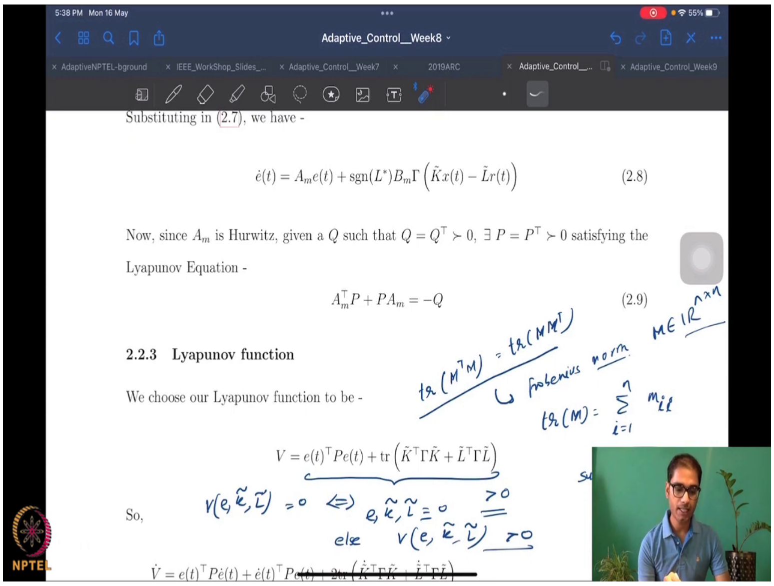


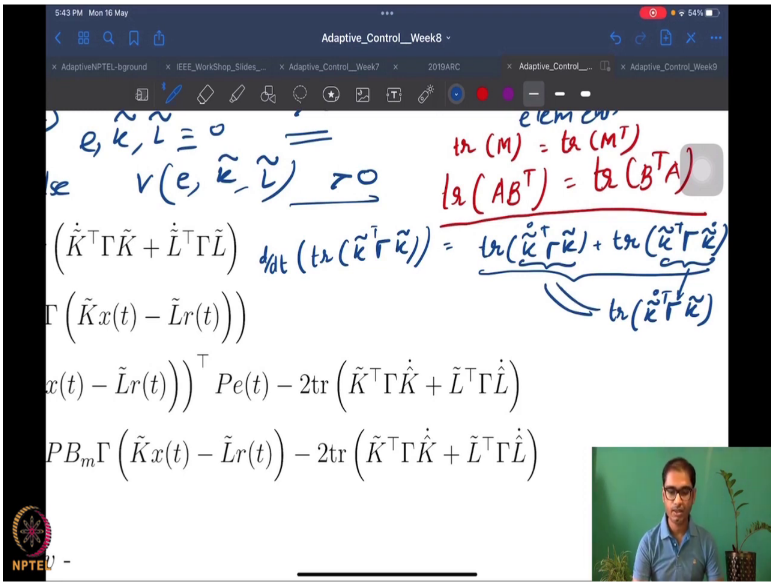
Now the difference, now that becomes apparent to us is that the parameters are not the scalars or vectors but in fact they are matrices. So, it is K tilde and L tilde and K star and L star which are all matrices we have already made an assumption that we know the sign of L star. So, L star is symmetric definite and we know whether it is positive definite or negative definite.

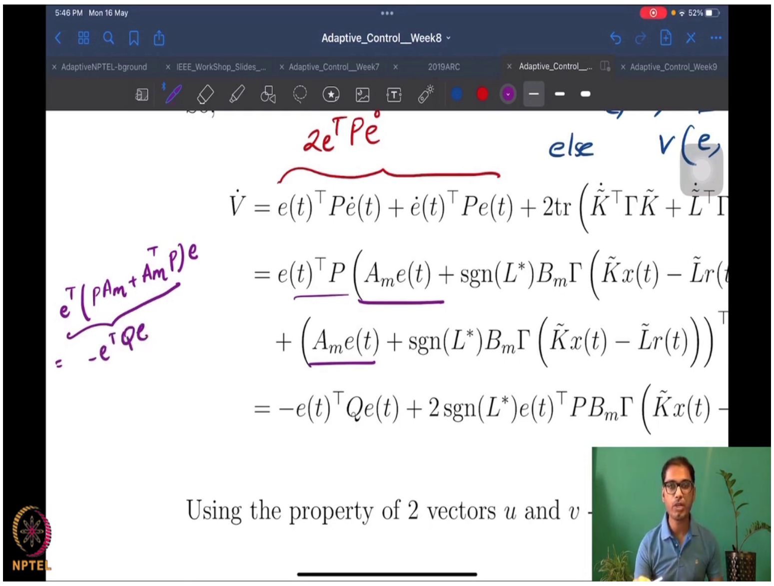
So, using this fact that we know the sign of L star we define a matrix gamma which is positive definite symmetric, which will appear sort of frequently in the Lyapunov analysis.

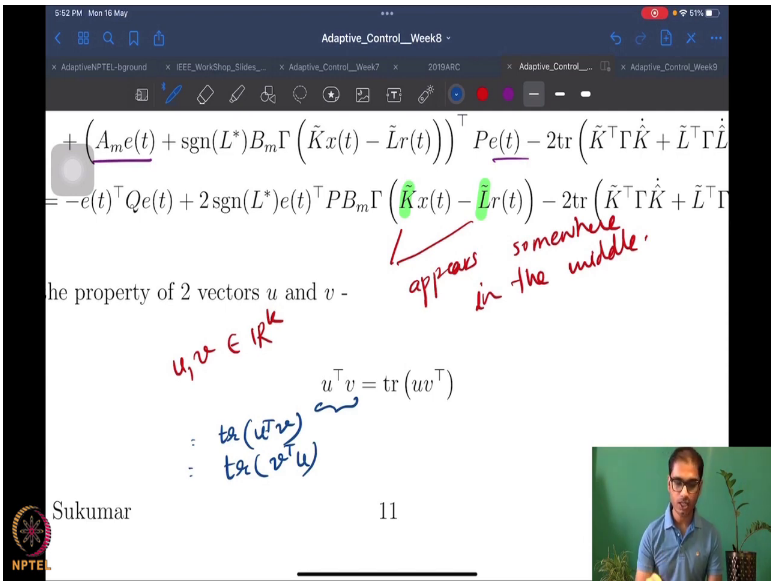
(Refer Slide Time: 3:09)

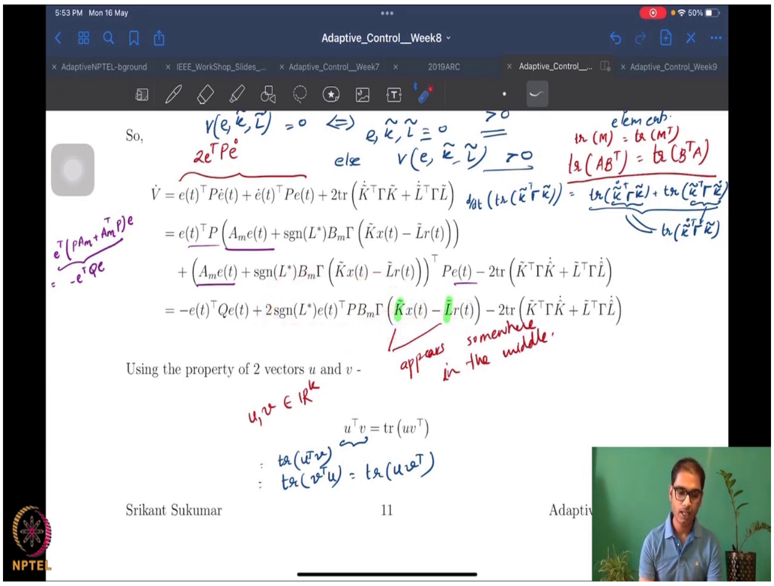


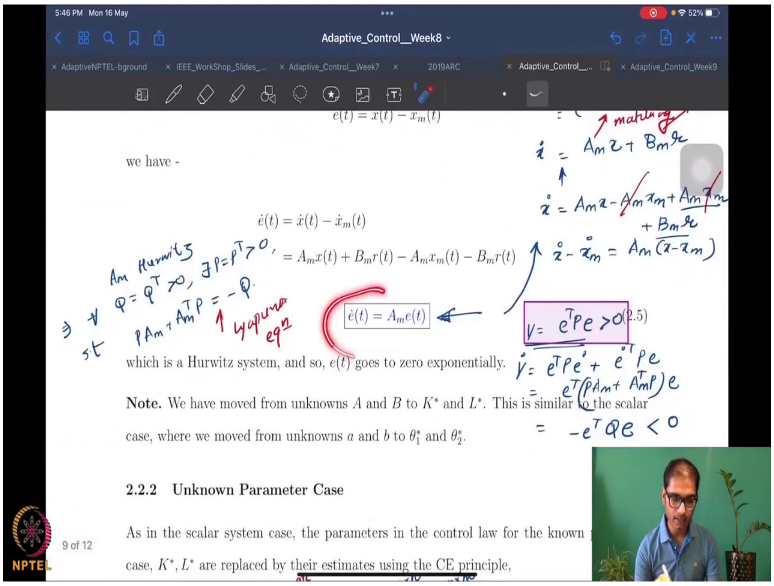



Adaptive_Control_Week8
$\tilde{e}, \tilde{k}, \tilde{L} = 0$
$V(e, \tilde{k}, \tilde{L}) > 0$
$tr(M) = tr(M^T)$
$tr(AB^T) = tr(B^TA)$
$\left(\dot{\tilde{K}}^\top \Gamma \tilde{K} + \dot{\tilde{L}}^\top \Gamma \tilde{L}\right)$
$\frac{d}{dt}\left(tr(\tilde{k}^T \Gamma \tilde{k})\right) = tr(\dot{\tilde{k}}^T \Gamma \tilde{k}) + tr(\tilde{k}^T \Gamma \dot{\tilde{k}})$
$tr(\dot{\tilde{k}}^T \Gamma \tilde{k})$
$\Gamma\left(\tilde{K}x(t) - \tilde{L}r(t)\right)\Big)$
$x(t) - \tilde{L}r(t)\Big)\Big)^\top Pe(t) - 2\mathrm{tr}\left(\tilde{K}^\top \Gamma \dot{\hat{K}} + \tilde{L}^\top \Gamma \dot{\hat{L}}\right)$
$PB_m\Gamma\left(\tilde{K}x(t) - \tilde{L}r(t)\right) - 2\mathrm{tr}\left(\tilde{K}^\top \Gamma \dot{\hat{K}} + \tilde{L}^\top \Gamma \dot{\hat{L}}\right)$
NPTEL


Adaptive_Control_Week8
$2e^T P\dot{e}$
else $V(e,$
$\dot{V} = e(t)^\top P\dot{e}(t) + \dot{e}(t)^\top Pe(t) + 2\mathrm{tr}\left(\dot{\tilde{K}}^\top \Gamma \tilde{K} + \dot{\tilde{L}}^\top \Gamma\right.$
$= e(t)^\top P\left(A_m e(t) + \mathrm{sgn}(L^*)B_m\Gamma\left(\tilde{K}x(t) - \tilde{L}r(t)\right.\right.$
$+ \left(A_m e(t) + \mathrm{sgn}(L^*)B_m\Gamma\left(\tilde{K}x(t) - \tilde{L}r(t)\right)\right)^\top$
$= -e(t)^\top Qe(t) + 2\,\mathrm{sgn}(L^*)e(t)^\top PB_m\Gamma\left(\tilde{K}x(t) -\right.$
$e^T(PA_m + A_m^T P)e$
$= -e^T Q e$
Using the property of 2 vectors u and v -
NPTEL


Adaptive_Control_Week8
$+\left(A_m e(t) + \mathrm{sgn}(L^*)B_m\Gamma\left(\tilde{K}x(t) - \tilde{L}r(t)\right)\right)^\top Pe(t) - 2\mathrm{tr}\left(\tilde{K}^\top \Gamma \dot{\hat{K}} + \tilde{L}^\top \Gamma \dot{\hat{L}}\right.$
$= -e(t)^\top Qe(t) + 2\,\mathrm{sgn}(L^*)e(t)^\top PB_m\Gamma\left(\tilde{K}x(t) - \tilde{L}r(t)\right) - 2\mathrm{tr}\left(\tilde{K}^\top \Gamma \dot{\hat{K}} + \tilde{L}^\top \Gamma\right.$
appears somewhere in the middle.
he property of 2 vectors u and v -
$u, v \in \mathbb{R}^k$
$u^\top v = \mathrm{tr}\left(uv^\top\right)$
$= tr(u^T v)$
$= tr(v^T u)$
Sukumar
11
NPTEL





So, using this gamma we now write our closed loop dynamical system in this particular form. So, this is the form that we would have obtained if there was a no parameter error or for the case when we do know the parameter. This is exactly what would be the closed loop dynamics but because we do not know the parameter there are terms in the parameter errors which is what you have commonly adaptive control.

And we choose a corresponding Lyapunov function using the fact that Am is a Hurwitz matrix and just like you would have used the norm a square norm for the vector case or just a square for the scalar case, we use a sort of norm a weighted norm for this matrix case also and this is defined via the trace of the matrix K tilde transpose gamma K tilde and similarly L tilde transpose gamma L delta. So, this is where we were.

So, I am going to mark the beginning here as lecture 8.2. So, I am going to mark this as lecture 8.2. So, this we already saw last time is in fact a validly Lyapunov function in fact it can be very easily claimed to be strictly positive definite also. So, no worries there we also know what is the trace operation it is essentially the sum of the diagonal elements of a matrix.

We also have certain nice properties of the trace like I mean it does not matter if you do the inner product or the outer product, the trace remains the same in fact does not have to be the same matrices either. So, you can change the sequence of things and the trace does not change and that is really what is important and we will in fact exploit this property.

So, one of the very very key properties of the trace like I said here but I can write it in a different color is the trace of A B transpose is equal to trace of B transpose A basically it is agnostic to the sequence in which I do things.

So, I can do A B transpose or you can do B transpose A and vice versa the trace is not going to change. So, this is one of the very very key properties that we will exploit soon. So, now let us take the derivative just like we do all let us diligently take the derivative and plug in the closed loop dynamics. So, we are yet to choose of course the update loss for the K tildes and L tilda’s and that is really what we are trying to do with this Lyapunov analysis.

So, V dot now remember these are vector operations we have to be very careful. So, because it is a vector. So, I take the derivative of all the terms very carefully. So, first I have e transpose P e dot basically taking the derivative of this, and then using the product rule I have to take the derivative of this term, so e dot transpose P e.

And now in this one I just say twice trace of derivatives here. Honestly speaking I could have written this also in this way, that is I could have written I mean let me actually sort of make this a little bit more clear. So, d dt of trace of say K tilde transpose gamma K tilde would have been written as trace of K tilde dot transpose gamma K tilde plus trace of K tilde transpose gamma K tilde dot, I could have done this I could have done this.

But the point is that because this entire thing is in fact a scalar. So, I am free to take a transpose I am free to take a transpose and of course transpose of I mean it is also true which I hope it is also evident to you that transpose of M sorry trace of M is equal to trace of M transpose. So, the trace does not modify under the transpose.

So, if I take the transpose here I would get if I take the transpose here in fact I would get trace of K tilde dot transpose gamma K tilde which is actually the same as this, which is actually the same of this. And that is why I am directly writing this as twice trace this. In fact I could have pretty much done the same thing here too let us be clear I could have done the same thing here too because P is a symmetric matrix.

So, I could have taken the transpose of this and I would have obtained e transpose P e dot which is the same as this but I am simply writing this separately because I am going to use the Lyapunov equation. That is it, that is the only reason I am not doing anything magical here. This could also be written as twice e transpose P e dot. Why?

Because this is a scalar, this is a scalar, transpose of a scalar is the same scalar. Therefore, I am free to take transpose of any term this is this we this is a trick we use regularly in all our Lyapunov analysis for vector equations. V is a scalar, so each term is a scalar. Therefore, I am free to take the transpose of any scalar term because it yields the same scalar term because I can take transpose of a scalar I end up with the same scalar nothing changes.

So, if I take the transpose of this term I get e transpose P transpose e. But because P is a symmetric matrix I get e transpose P e dot. So, e transpose P transpose e dot is the same as e transpose P e dot which is the same as this term. So, I could have written this very well as twice e transpose P e dot I simply am not choosing not to write it in that way because I want to write the Lyapunov equation or use the Lyapunov equation.

So, what is this? So, when I substitute for e dot now in on both sides here, so this remains as it is for a while, this remains as it is I just use the fact that K tilde dot is equal to minus K hat dot and that is what shows up here and that is it. So, again I have flipped it here, this is flipped here, and this is also flipped here do not mind I can flip as many times as I want it does not change anything because trace of A is equal, trace of M is equal to trace of M transpose.

So, these terms remain as it is just with the negative sign because the K tilde dot has been changed to K hat dot. And similarly, L tilde dot has been changed to L hat dot that is what is bringing up the negative sign. So, so let us see, so then what do we do, we basically plug in for the dynamics e dot that is a closed loop dynamics e dot from here, that is what you see here, that is it.

It is Am e plus sgn L star Bm gamma this thing and similarly here with the transpose this guy. Now these terms and this term gets combined to give me this term this is just the Lyapunov equation. So, if you may, this term, this term with this term, whatever gets combined and can be written as e transpose P Am plus Am transpose P e and that is actually equal to minus e transpose Q e from the Lyapunov equation corresponding to the Am, which is a Hurwitz or stable matrix.

Now, so that is how I get this nice negative term this is the nice negative definite term that we are always looking out for in the Lyapunov analysis. So, we already have that term and why this is because we chose a nice target system. So, we chose this as a target system and that is why we get this nice negative term and Am is a Hurwitz matrix.

So, then we if you look at the rest of the terms all the terms that remain which is these guys, these guys these are all parameter error terms. So, again this term and this term can also be combined similarly this term and this term can also be combined and that is what is being done here. These are combined to write them as two sgn L start e transpose P Bm gamma and this.

Both these terms are exactly the same, both these terms are, again using the logic that V is a scalar, so every term is a scalar therefore I can take transposes as many times as I want. So, so this term combines with this term to give us this two times term. And we also have the same two coefficient here as you would expect and in this form.

Now if you see we have a little bit of a funny situation here, earlier when we were doing all this parameter update law design, we would always get a theta tilde common on the left hand side and all that because it was a scalar, so we could move the theta tilde anywhere. Now the problem that we have here is that the K tilde which we want to take common outside is somewhere in the middle and these are all matrices.

So, it is not like I can move it to the left and right arbitrarily. So, this is important to note. So, this term and this term. So, so appears somewhere in the middle we would ideally want it to be on the left or the hand side. So, that we can take things common and whatever cancel it out using K hat dot this is what we have been doing, now what do we do, well, we use nice tricks from the trace properties.

So, this this is where we see one another trace property. So, we already know this that trace of M is the trace of M transpose and trace of A B transpose is trace of B transpose A. In fact from these two I can derive this. So, if you, so here we are saying that trace of u transpose v where both u and v are vectors are vectors is the same as trace of u v transpose.

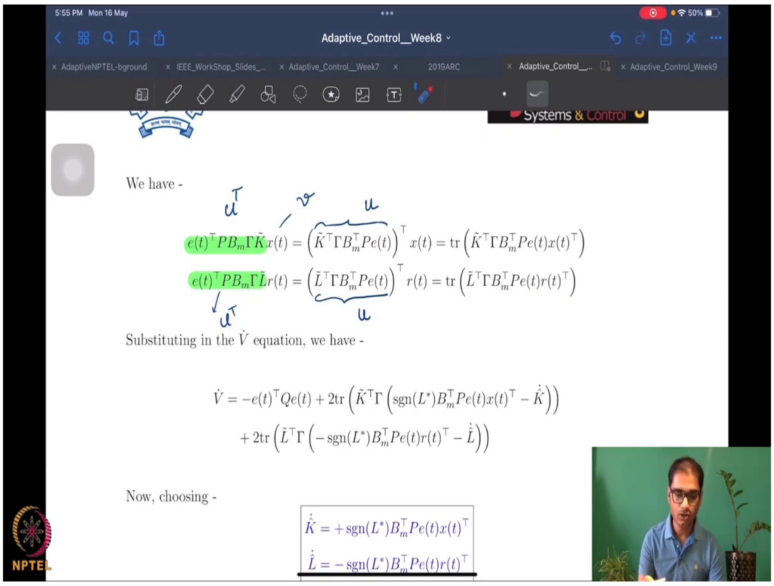
So, basically if you have vectors of the same dimension, so u v has to be vectors in some R k has to be the same dimension otherwise inner product is not different. And you take the inner product that is u transpose v then this is the same as trace of the outer product. So, notice that this is a scalar and this is an n by n matrix or a k by k matrix and because I take a trace it becomes a scalar. So, both sides are scalar.

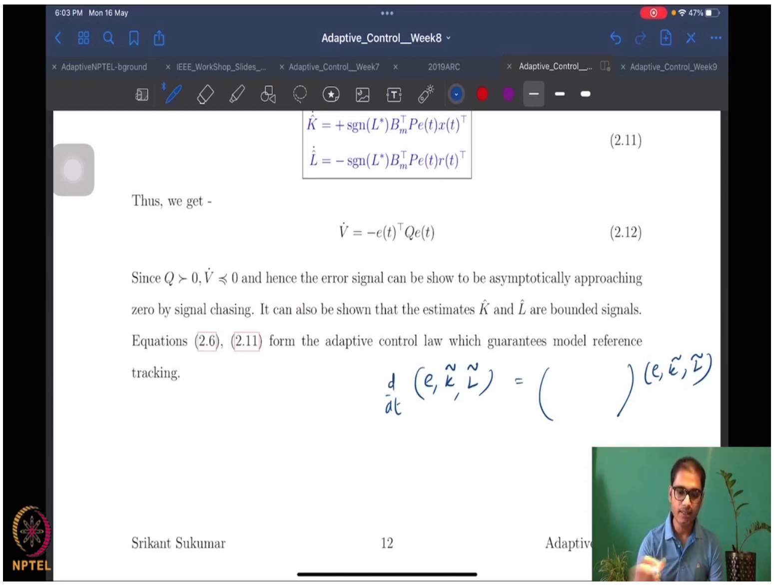
So, this one needs to verify of course for our own sanity and then we can try to use these results let us try I do not know I mean maybe we maybe it works maybe it does not work. So, so let us see. So, this is actually equal to trace of u transpose v it is a scalar. So, this is also the same as trace of v transpose u because that is the same scalar take a transpose of a scalar I say that get the same scalar and the trace of a scalar is also the same scalar because scalar can be seen as a trivial matrix of a 1 by 1 matrix.

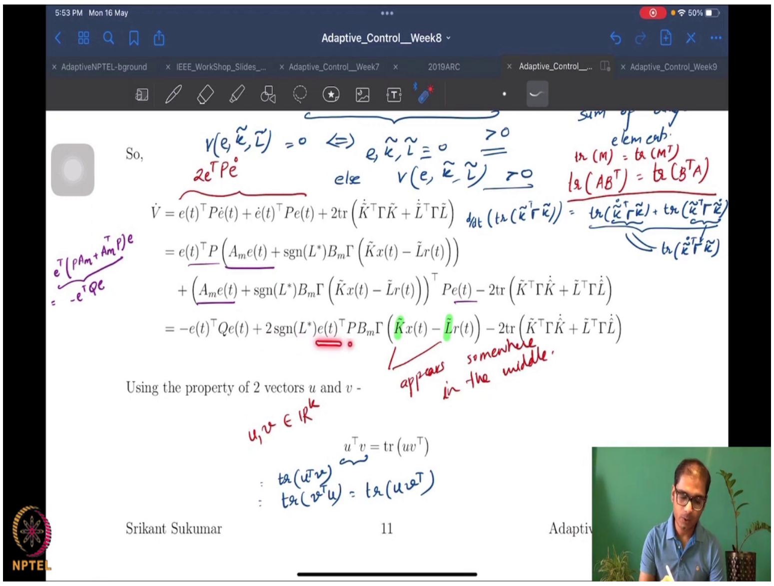
So, the sum of diagonals is just the scalar itself. So, u transpose v is the same as trace u transpose v and trace u transpose v is the same as trace v transpose u because again we are just transposing it using this identity. And then finally I can flip the order I am allowed to flip the order of the operation with the trace this is the from this property. So, this this whatever this property that you have can be obtained from the other properties that we already had.

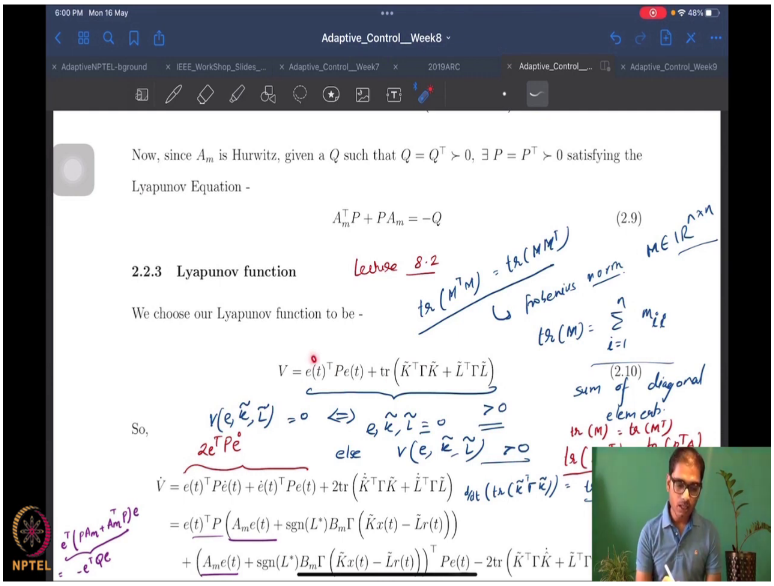
So, this is not a new property per se, it is in fact just something that is derived from the existing property. So, these two are the most key ones if you remember these you can mostly derive the other important trace equalities. So, now we have this very nice trace equality that the inner product is equal to the trace of the outer product of vectors and that is what we are going to apply to these terms this term and then this term.

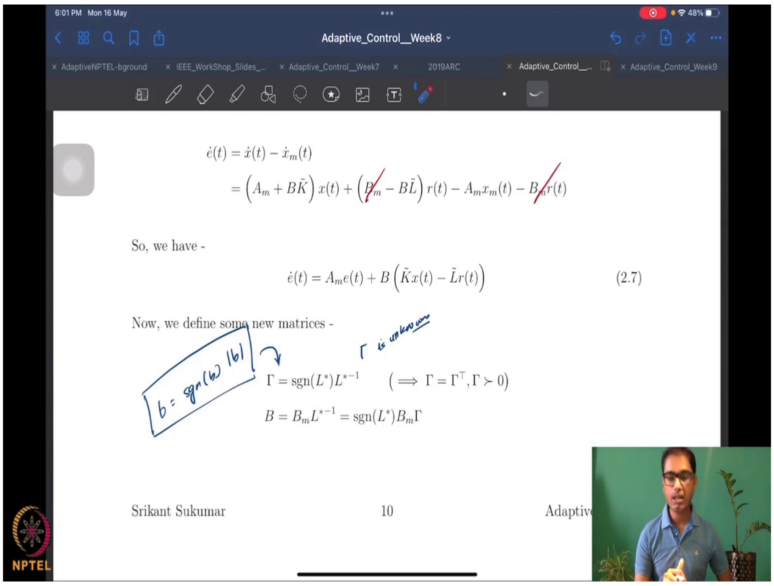
(Refer Slide Time: 19:01)


Adaptive_Control__Week8
We have -
$e(t)^\top PB_m\Gamma\tilde{K}x(t) = \left(\tilde{K}^\top\Gamma B_m^\top Pe(t)\right)^\top x(t) = \mathrm{tr}\left(\tilde{K}^\top\Gamma B_m^\top Pe(t)x(t)^\top\right)$
$e(t)^\top PB_m\Gamma\tilde{L}r(t) = \left(\tilde{L}^\top\Gamma B_m^\top Pe(t)\right)^\top r(t) = \mathrm{tr}\left(\tilde{L}^\top\Gamma B_m^\top Pe(t)r(t)^\top\right)$
Substituting in the $\dot{V}$ equation, we have -
$\dot{V} = -e(t)^\top Qe(t) + 2\mathrm{tr}\left(\tilde{K}^\top\Gamma\left(\mathrm{sgn}(L^*)B_m^\top Pe(t)x(t)^\top - \dot{\hat{K}}\right)\right)$
$+ 2\mathrm{tr}\left(\tilde{L}^\top\Gamma\left(-\mathrm{sgn}(L^*)B_m^\top Pe(t)r(t)^\top - \dot{\hat{L}}\right)\right)$
Now, choosing -
$\dot{\hat{K}} = +\mathrm{sgn}(L^*)B_m^\top Pe(t)x(t)^\top$
$\dot{\hat{L}} = -\mathrm{sgn}(L^*)B_m^\top Pe(t)r(t)^\top$
NPTEL


Adaptive_Control__Week8
$\dot{\hat{K}} = +\mathrm{sgn}(L^*)B_m^\top Pe(t)x(t)^\top$
$\dot{\hat{L}} = -\mathrm{sgn}(L^*)B_m^\top Pe(t)r(t)^\top$ (2.11)
Thus, we get -
$\dot{V} = -e(t)^\top Qe(t)$ (2.12)
Since $Q \succ 0, \dot{V} \preccurlyeq 0$ and hence the error signal can be show to be asymptotically approaching zero by signal chasing. It can also be shown that the estimates $\hat{K}$ and $\hat{L}$ are bounded signals.
Equations (2.6), (2.11) form the adaptive control law which guarantees model reference tracking.
$\frac{d}{dt}(e, \tilde{k}, \tilde{L}) = (\quad)(e, \tilde{k}, \tilde{L})$
Srikant Sukumar 12 Adaptive
NPTEL


Adaptive_Control__Week8
So,
$V(e, \tilde{k}, \tilde{L}) = 0 \iff e, \tilde{k}, \tilde{L} \equiv 0$
else $V(e, \tilde{k}, \tilde{L}) > 0$
$\mathrm{tr}(M) = \mathrm{tr}(M^T)$
$\mathrm{tr}(AB^T) = \mathrm{tr}(B^TA)$
$\dot{V} = e(t)^\top P\dot{e}(t) + \dot{e}(t)^\top Pe(t) + 2\mathrm{tr}\left(\dot{\tilde{K}}^\top\Gamma\tilde{K} + \dot{\tilde{L}}^\top\Gamma\tilde{L}\right)$
$= e(t)^\top P\left(A_m e(t) + \mathrm{sgn}(L^*)B_m\Gamma\left(\tilde{K}x(t) - \tilde{L}r(t)\right)\right)$
$+ \left(A_m e(t) + \mathrm{sgn}(L^*)B_m\Gamma\left(\tilde{K}x(t) - \tilde{L}r(t)\right)\right)^\top Pe(t) - 2\mathrm{tr}\left(\tilde{K}^\top\Gamma\dot{\hat{K}} + \tilde{L}^\top\Gamma\dot{\hat{L}}\right)$
$= -e(t)^\top Qe(t) + 2\,\mathrm{sgn}(L^*)e(t)^\top PB_m\Gamma\left(\tilde{K}x(t) - \tilde{L}r(t)\right) - 2\mathrm{tr}\left(\tilde{K}^\top\Gamma\dot{\hat{K}} + \tilde{L}^\top\Gamma\dot{\hat{L}}\right)$
appears somewhere in the middle.
Using the property of 2 vectors u and v -
$u, v \in \mathbb{R}^k$
$u^\top v = \mathrm{tr}(uv^\top)$
$\mathrm{tr}(u^Tv)$
$\mathrm{tr}(v^Tu) = \mathrm{tr}(uv^T)$
Srikant Sukumar 11 Adaptive
NPTEL





So, let us look at each of them. So, this term the first term. So, we do not write the sgn L star and all that but you look at the rest of the term. So, basically we ignore this two sgn L star and you look at e transpose P Bm gamma K tilde x, e transpose P Bm gamma K tilde x. So, if you look at this term I know this is a scalar. So, what do I do I think of this whole piece. So, this entire piece until here I think of as my u transpose and this is v.

So, basically this is of course u because I took the transpose and that is v and I know that the inner product here is the same as trace of the outer product which is u v transpose. So, what do I write it as I just write this as u. So, this whole thing is u, it is K tilde transpose gamma Bm transpose P e and so this is the u and then this is V transpose. So, I get something like this.

We do the same thing on the other term that is also a similar term it is if I ignore this because this is just a scalar and can be moved around. So, I do not need to worry about this. I look at the second term which is e transpose P Bm gamma L tilde r e transpose P B m gamma L tilde r. So, similarly I think of this term everything before the r everything before the r is my u transpose.

So, again this is u and so this is u transpose v and the this is essentially trace of u v transpose. So, what does this do notice in this expression K tilde is somewhere in between of no use to us but in this new expression K tilde is all the way to the left and the trace appears. So, earlier there was no trace either. So, and why do we need the trace?

Because the comparison term in K tilde contains the trace or is within the trace. So, obviously I need somehow the trace function to appear here and I also want the K tilde to be somewhere in the left because here the K tilde appears in the left. Similarly, for the L tilde I here it is somewhere in the middle with no trace.

So, I want to this to appear in the left inside the trace function and that is exactly what happens this appears inside a trace function there is a K tilde transpose in the beginning and then again a trace function and then L tilde transpose in the beginning. So, we essentially substitute back into this this back into the V dot equation I am again left with this nice negative definite term then I have a twice trace sgn L star is remaining as it is.

And then I have taken because this is a scalar I can move it around wherever I want. So, I move it inside here then I take the K tilde gamma out. So, I get a sgn L star here from this term sgn L star Bm transpose P e x transpose. So, I cannot change the sequence of the matrices. So, this is fine and notice there is a K tilde transpose gamma here, which I take common K tilde transpose gamma here that is taken common.

So, I am left with this guy sgn L star and this and a K hat dot here. Similarly, there is an notice that there is a L tilde gamma L tilde transpose gamma here common. So, again L tilde transpose gamma inside the trace common that is what I take out and then the sgn star is here of course that is a negative sign and then I am left with this term and then L hat dot.

So, this is just very careful bookkeeping now once I have applied this nice trick which gives me a trace and also pulls the K tilde and L tilde to the left most I can now take common with the terms here where the K tilde and the L tilde are in the left in fact I take both K tilde and times gamma common and L tilde transpose times gamma common.

And once I have this I am in good shape because now I can use my K hat dot to cancel this guy similarly L hat dot to cancel this guy which is exactly these equations brings about of course the sgn L star. So, which is what we have already discussed that we do need to know the sign of L star. So, we need to know whether it is positive definite or negative definite this condition or this requirement is exactly identical to knowing the sign of B in the scalar case nothing this is something we cannot avoid.

So, then you basically get rid of these terms and all you are left with is this guy and we know that this is because Q is positive definite V dot is negative semi-definite not definite because we have additional states K tilde and L delta. So, obviously it is not negative definite it is in fact negative semi definite.

So, what can we show? We can show that the error signal e goes to 0 asymptotically as t goes to infinity this we can do by signal chasing and using Barbalat's lemma. Of course also you can show that these are bounded signals no problem. Why? Because simply because V was chosen as a positive definite function containing both e quadratic essentially some kind of quadratic in e K tilde and L tilde and we have also shown that V dot is less than equal to zero. So, V is non-increasing function of time.

So, as time progresses V is non-increasing, which means that whatever is the initial value of V you cannot actually go beyond it therefore these things cannot go unbounded none of these terms can go unbounded and if none of these terms can go unbounded they are essentially norms which means e K tilde L tilde cannot go unbounded.

So, notice that we have used gamma in the Lyapunov analysis but gamma does not appear anywhere in the update law. It cannot because gamma is unknown although we constructed this gamma using this using this L star inverse this is unknown therefore gamma cannot appear anywhere in the update law. So, so again, so this is also again similar to this absolute value of B that we introduced in the Lyapunov function for the scalar case.

So, this is again something similar to that. So, so that is it. So, we are done we essentially have a nice model reference adaptive controller. This is what guarantees tracking of course it does not guarantee any kind of parameter identification again this is again I would say this is something that will that requires persistence of excitation.

And in fact I strongly recommend that you write the if you want to write the dynamical system in a very careful form with e K tilde K tilde and L tilde as states if you write the dynamical system in this form with some matrix. So, d dt with some matrix e but I am writing it as a tuple (specif) very deliberately because K tilde and L tilde are matrices.

And e is a vector therefore you cannot actually write it in this form you have to vectorize K tilde and L tilde but if you do write it you will get appropriate conditions for persistence of excitation of a particular signal which will guarantee that parameters also converge asymptotically. So, this anyway we have seen this theory I would recommend that you write it in this form and see how it looks all but I mean we have perfect tracking which is what we typically guarantee in adaptive control anyway.

So, this was a model reference adaptive control. So, what we did today was complete the Lyapunov analysis for the model of preference adaptive control problem. And in the process of the Lyapunov analysis we learned a lot about trace and trace properties and how they can be used to construct a Lyapunov functions for matrix states.

We also learned about trace properties and how it can help us to manipulate the matrices that appear and move matrices from one location to another. So, that we can cancel some terms. So, we learned some interesting trace properties which are also very useful for a lot of different mathematical analysis and of course we were able to prescribe update loss for these matrix unknowns K tilde K star and L star.

And we were able to prove perfect tracking which is what we typically seek to guarantee in model reference adaptive control. So, that is sort of the exposition into model reference adaptive control for linear systems. We will now subsequently move into a new and different set of lectures where we will discuss some other and more novel topics. So, this is where we stop, thanks and I hope to see you again soon.