Transcribers Name: Crescendo Transcription Pvt. Ltd.

Nonlinear Adaptive Control
Professor Srikant Sukumar
Systems and Control
Indian Institute of Technology Bombay
Week 2
Lecture 8
Barbalat's Lemma Part 2

Hello everyone, welcome to yet another session of our NPTEL on Nonlinear and adaptive control, I am Srikant Sukumar from Systems and Control, IIT Bombay. We are again back to our motivating image, we are in week 2 and we aim to sort of algorithms and analyze them and develop algorithms that will help us drive autonomous systems such as this.

So, we go back and try to recap what we did last time before moving ahead very quickly. So, last time we started by studying a few important lemmas, the first one being on convergence of certain functions. And in this Lemma, we saw that if you have a function which is scalar valued and with the derivative which is bounded then the function is uniformly continuous.

This was one of the Lemma and this was one of the Lemma the other Lemma we saw in fact, we saw this Lemma 1.1 earlier which said that if you have a scalar valued function again and it is bounded below and non-increasing, then the function has a finite limit as t goes to infinity. You also saw examples of what such functions are.

We then studied one of the at least one version or the one version of the Barbalat's Lemma, which is one of the very very key Lemmas for analysis of adaptive systems without Barbalat's Lemma, there is almost no hope of doing any adaptive systems analysis and here in this Lemma essentially, we say that if a function is integrable and here we could talk about both scalar and vector valued functions, for vector valued functions we look at component wise integrals.

So, if the function is integrable and further that the function is uniformly continuous then we have that the function converges to 0 as t goes to infinity. We also saw an example of how this Barbalat's Lemma although it is the sufficiency condition that is it is a one way result it says this if these 2 conditions are satisfied, that is integrability and uniform continuity then you have a convergence to 0 and not the other way around not the other way around.

So, this is a sufficiency condition, but we saw an example of how this is a rather tight sufficiency condition, we saw an example where the function is integrable but not uniformly continuous, then even in this case, the function does not actually converge to 0, we saw that there is in fact no limit to the function of this kind.

So, moving on, we will look at a corollary of this. We will look at the corollary of this. And we want to say that, we want to sort of look at a slightly simpler looking condition if I may. And there is a more concise statement if you but still remember, this is a corollary. So, what does it say? It says that if the function is such that it is both L infinity and L p for some integer p and where infinity is not included because of course, you already have infinity here. So, if it is L infinity and L p for some p and further f dot is L infinity, then the function converges to 0 as t goes to infinity.

So, this is sort of a, if I may sort of a different characterization for Barbalat's Lemma. Now, like I said it is a corollary and so it makes sense to sort of want to know that corollary implies the actual Lemma itself, and so therefore or we do say something like this, we said that if the function is L infinity and L 2 then we want to prove that limit as t goes to infinity f t is equal to 0. So, we essentially want to prove the corollary.

So, this is I would say you also want to add here and also want to add here that the derivative is infinity. So, we want to prove that the corollary holds true when this p is equal to 2. So, this is sort of a specific case of the corollary but if you can prove it for p equal to 2 you can prove it for any p no problem and so, this is sort of asking us to prove the corollary using original Barbalat's Lemma.

So, you want to prove the corollary using the original Barbalat's Lemma statement, which is essentially Lemma 2.1. So, I leave it to you and this will be part of the homework exercises of course. So, I want to, want you to give this as a shot. If you remember I already said that this integrability condition has some similarity with L1 type condition. So, and you sort of have to use similar idea here. That is sort of the hint for this exercise.

Now, let's see how one might use the Barbalat's Lemma I want to give you a glimpse of convergence analysis using this very, very cool tool that we are claiming can do so much. So, you want to see if it can in fact, what we are seeing it can and will help us analyze convergence of adaptive systems.

So, suppose we consider this typical spring mass damper type system you all have seen something like this before in most of your mechanics dynamics in high school and so on. The dynamics of this system is given by this equation this is very, this equation 3.1 is very standard application of the Newton's second law. I mean, what you would typically do is write think of M as a freebody diagram.

So, the M as a free body and then we write all the forces on it. So, what are the forces you have something like let's see, there is a sort of inconsistency here and this should be so, let me change this this should be c and this should be c and not b. So, this there is this is a free body there are multiple forces here.

So, this movement is only horizontal. So, let me try to write this a little bit carefully. So, this is the vertical force is mg. And then you have a normal reaction from the ground. I mean, it is not really marked here, but there is sort of a constraint here. To prevent it from falling below. So, that is a normal force and then in this direction, we have forces C x dot due to the damper and you have kx. You have C x dot to the damper and you have kx due to the spring.

So, again this x is of course, this x depends on this displacement. Make sense? And then of course, had this user applied force f of t and then what does the Newton's law state, it states that the, if I write it in only the horizontal direction in the x direction, the Newton's law will be mx double dot is equal to all the external forces.

So, the only positive force is f of t and then you have minus C x dot and minus kx. So, once I have written this very, very concisely the Newton's law, let us see, let us see, I want to sort of, so, this concise form of Newton's law has been written and see, let me see if I can make this bigger. looks like I can, so, once you have written this very concise form of the Newton's law, you can see that this is exactly what we have here is exactly the same equation, just rearranging the terms here.

So, now that I have this equation, I am going to write it in our standard state space form, like and how do I write it in a state space form by choosing states? So, what are my states, I choose my states as x 1 is equal to x and x 2 is equal to the derivative of x, this is the standard way on how you convert anything into a state space form.

So, then what do we have we have x 1 dot is equal to x dot, which is actually equal to x 2, so, that is the first equation, then x 2 dot is the second derivative of x and this can immediately be concluded from this equation. It is pretty obvious. So, these 2 equations are in fact the same. So, here I have mx double dot, which is essentially mx 2 dot and then you have cx dot which is just sorry, I am sorry that is a small letter here this is not states are like this x 1 is x and x 2 is x dot.

So, x 1 dot is x 2 and then x 2 dot is x double dot, which is minus c by m x dot which is x to minus k by m x, which is x 1 plus f over m. That is f over m. So, now, in general as it is mentioned here, in adaptive control as well as in nonlinear control, we typically want a state to follow desired trajectory, may seem rather unrealistic to you for this particular problem, but that is what we always want to do.

We want our robots to follow a trajectory we want our spacecraft to follow a trajectory you want our voltage signals to follow a trajectory and so on and so forth. I mean, if you consider any system you in general want your state to follow a trajectory. And therefore, we define this desired trajectory.

So, this is we define what is called desired states by just these x 1 desired and x 2 desired usually, you will have to have the x 2 desired to be the derivative of x 1 this is so, that this has to be consistent with the dynamics this is sort of a matching condition the fact that x 1 desired and x 2 desired are related in this way is a matching condition.

Why is this a matching condition? This is because, if you see for this dynamical system itself x 1 derivative is actually equal to x 2 in this particular case, x is x 1 is the position and x 2 is the velocity. So, the position derivative is the velocity. Therefore, in order to have a consistent desired trajectory, you will need your desired trajectory to also have the derivative of position to be the velocity of the desired trajectory.

Otherwise these are not compatible trajectories and you cannot track such a trajectory. If I give you a trajectory where x 1 dot desired is not equal to x 2 desired then this trajectory cannot be followed by the system, this should be obvious to you. So, that is why this is called a matching condition, so this makes sense. That if your system has position derivative equal to velocity, then your trajectory also has to position derivative equal to velocity. There is no 2 ways about it.

Now, that I have these desired trajectories, what do I do? As control engineers, we are always interested in looking at things going to 0. Remember, we always said that our equilibrium we assumed to be at the origin and so on and so forth. So, we are always interested in going to 0 if you see the Barbalat's Lemma, is also saying that signal goes to 0.

So, because we want to deal with objects that go to 0, and we want to create error variables, , because I do not want to look at x 1 going to x 1 desired, because both x 1 and x 1 desired quantities that depend on time. So, what do I do I construct an error, which is the difference of x 1 and x 1 desired, this is very standard. And similarly, an error 2 that is e 2, which is the difference of x 2 and x 2 desired. So, I create these errors corresponding to created corresponding to all system states, we create this corresponding to all the system states.

So, here we have 2 states, so, we create 2, if you are 100, we will create 100 such errors. Why does this make sense? Why does this make sense? Why it makes sense is because, if this is equal to 0, if this is going to 0, if both these errors are going to 0, what do I have? I have what I desire, I have this implies that x 1 goes to x 1d and x 2 goes to x 2d. And this is precisely my tracking objective.

This is precisely my tracking objective that I want my desired quantities to go to the true values. I hope this makes sense to all of you, because this has to make sense to you, because this is what we do in all of nonlinear control and adaptive control, we always create error variables or variables, which we want to go to 0.

We always want to create variables that will go to 0 and therefore, we create error variables. Instead of looking at the original x 1 x 2 variables, we look at x 1 minus x 1 desired and x 2 minus x 2 desired as my new system variables. So, once I have these error variables, these new definitions. What is the d? I want to identify the dynamics or the evolution of these other variables, until now I had dynamics of x 1 and x 2.

But now I want dynamics of e 1 and e 2, and how do I do then, just take the derivative because I have the dynamics of x 1 x 2 and if I want the dynamics of e1, e 2, so I just take the derivative of e 1 and e 2, that is what I do here e1 dot is just x 1 dot minus x 1 desired dot, this and what is x 1 dot minus x 1 desired dot? It is exactly same as our definition of e 2.

And this is not by magic, this is not by magic on by any distance. This is by virtue of the matching condition. This because of the matching condition that you have this sort of dynamics. So, what happens e 1 dot turns out to be exactly e 2. So, aerodynamics, the first equation of aerodynamics looks exactly identical to our original state dynamics. It is an integrator, it is an integrator, it is an integrator, it is very standard in mechanical system and aero mechanical systems at least. You have an integrator as your first state dynamics.

Then what about the second state you have e 2 dot and then you take derivative of x 2 and x 2 desired. So, you have x 2 dot and x 2 desired dot, x 2 dot you can plug in from my equation such as this and this is the entire equation that we had from before and you have x 2 dot which is again the double derivative of x 1, just the double derivative of x 1. As of now we are assuming I mean, we are still not in the purview of any adaptive control or anything we are not even learned what is adaptive control? So, we are simply trying to understand what you know, Barbalat's Lemma can do for us.

How do you analyze systems using the Barbalat's Lemma. So, as of now, we are simply trying to do that and just to give you a taste of how analysis with Barbalat's Lemma goes. So, we assume for the moment that all the parameters of the system are known that is we know this, we know this quantity, we know this quantity. So, all these parameters are assumed to be known.

Obviously, the trajectories are have to be known have to be given to us. Dosen't make sense otherwise? Because, if I do not know what I am following then, what am I following. So, what do I do? I start off by choosing an appropriate controller. Now, what is this appropriate controller. I know for a fact that well I mean, maybe you don't, but I know, for a fact that a system like 3.6. There is this guy is exponentially stable system. You already know what is the meaning of this.

So, this is a globally exponentially stable system. I know this for a fact. Why do I know this for a fact? Well, because I am doing control engineering for a long time. So, I know this for a fact. Now, for those of you who do not know this for a fact, how can one verify that such a system is exponentially globally, exponentially stable in many many different ways. One one obvious way is just this is a linear system first we identify that this is a linear system.

And then what do you do you say e is e 1, e 2 transpose, so, then you have e dot is 0 1 minus k 1 minus k 2 e and then what do I do? I compute eigenvalues you can do many things once you do this, you can once you have this sort of equation, you can do many things you can write the characteristic polynomial and use the Routh Hurwitz criteria, any of you might have done it in your typical first basic control systems course.

But, I mean in this case, I can also compute the Eigen values directly. One way or another it is not difficult to see like what is what will be the, what will be the characteristic polynomial of the system, it will be something like lambda square plus k 1 s plus k 2 equal to 0. Let is see, let me sort of verify this is going to be the case. So, this is I am claiming the characteristic polynomial.

What is the characteristic polynomial? It is simply the determinant of sI minus A equated to 0. This is a well I mean, not s let us use Lambda, I minus A equated to 0. I am simply computing the determinant of Lambda I minus A. So, this is just determinant of Lambda minus 1 k 1 lambda plus k 2 and that is what I am equating to 0.

So, actually this is the other way around let me see. So, you will get lambda squared plus k 2 lambda plus k 1 this is what you will get, let me sort of highlight this nicely, this is what you get as your characteristic polynomial and it is not very difficult to see using this particular characteristic polynomial that what is going to be your Eigen values.

It is simply the solution of this, it is simply the solution of this, so, what is the solution? Let us see, I mean, lambda is going to be equal to. So, let me write here the left, lambda is going to be I apologize. Lambda is going to be equal to minus k 2 plus minus, it is gonna be minus k 2 plus minus square root of k 2 squared minus 4 k 1 divided by 2.

And depending on what this k 2 and k 1 are, you will get different roots. But the key thing to remember is that, this is always, the key thing to remember, is that real part of lamda is always negative here. Why would the real part of lambda always be negative? I know that because this quantity in the bracket here that is this guy is always less than this guy.

This is always less than this. So, because this is less than this, the real part of lambda will always be negative. Again, this is something you know, from your standard first, you know, first level course in control system, so therefore, I am not covering it in too much detail. But you should know that, if the quantity in the square root is less than, this quantity here, then the real part has to be negative.

Real part has to be negative, and the real part is negative. I know, for both the eigenvalues, then I know that this system is globally exponentially stable, or of course, like I said, you can simply use the Routh criteria, and you know that k 1 and k 2 are both positive. And you can continue along and it is very easy to conclude that the system is exponent globally, exponentially stable.

So, now cool. So, I know that this equation in blue is globally exponentially stable. So, what do I want? I want this system to follow this system that is to be the same as this system. Because if I can make 3.4 and 3.6 same, then I am done. Because my system is globally exponentially stable. So, how do I do that? Because I know, why do I use this target system, because I already see that the first equation is matching.

If the first equation was not matching, then I had no hope, because I have control. That's it this f over m is what is my control? The control function is f is the control function. So, I have control on the second equation, but nothing in the first equation. So, the first equation definitely had to match and they do.

So, I choose this target system. And so, what do I do? I choose this f over m in a way that my hand side becomes this and that is exactly what this is, I have simply chosen to cancel this guy. This, this and this and that's it, I have simply chosen to cancel these quantities. Which are new definition here, of course, I am calling this key 1 as k by m, k 2 as c by m and then I introduced some nice terms, minus k 1, e1 minus k 2 e 2 the design introduce these terms here.

Basically this, I get from here to here, by choosing the f. That is, that is the end and I do that. And so, I get that my, what is called my closed loop, this is, the what I would call not just error dynamics, but closed loop error dynamics that is I have plugged in the control for it is called the closed loop error dynamics.

Now, of course, we know that this system using our either Routh-Hurwitz methods or using our using eigenvalue analysis that this is exponentially stable, but what if I want to prove this using the Lyapnov methods or using a potential function this is what we will see next time it can be done using Barbalat's Lemma.

Why do we want to do this, is because typically nonlinear systems may not have you know such easy nice structure you may have something very nonlinear appearing here and then you cannot use route criterion and so on and so forth, your target system may not be linear, and in such cases you are forced to use nonlinear analysis methods. And that is what we will see how to prove exponential convergence using Barbalat's Lemma and Potential function methods, next time.

So, what did we do? This time, we essentially started to look at how to use the Barbalat's Lemma. We are not there yet. We will continue of course next time. But we have sort of looked at a model, looked at a control and looked at an error equation. Before that, we saw an alternative version, or a corollary to the Barbalat's Lemma. Which also may be useful to us at some junctures. So, this is where we will stop today. See you again next time. Thanks.