Transcriber's Name: Crescendo Transcription Pvt. Ltd.
Nonlinear Adaptive Control
Professor Srikant Sukumar
Systems and Control
Indian Institute of Technology, Bombay
Week 6
Lecture No: 35
Adaptive Control Design: Second Order Systems

Hello everyone, and welcome to yet another session of our NPTEL Non Linear and Adaptive Control, I am Srikant Sukumar from Systems and Control, IIT Bombay. As always, we have this nice background image of the SpaceX satellite in our background. And we are hopeful that it motivates us to develop the design algorithms that help systems such as these operate autonomously. So, in the sixth week, we have been, we have started our excursion into real adaptive control problems and adaptive control design.

And we have already covered and adaptive control design for first order scalar system. That is what we did, we also analyzed the stability. And we could essentially conclude that we can show that all closed loop signals are uniformly stable with respect to the origin equilibrium. And we could also show that the tracking errors converge asymptotically to 0.

What we could not guarantee was that the parameter estimation errors, in fact goes to 0. And for this, we said that there is requirement for persistence of excitation condition. And this is what we had looked at in the week proceeding to this. And we sort of used a little bit of that theory that we learned in order to claim parameter convergence for certain special case scenarios.

So this was what we had for the first time order system. At the end of the last session, we sort of started looking at a second order scalar system, that is not significantly complicated. What we do here is just add an integrator to the standard first order dynamical system. And, as usual, our, we have x1 and x2 evolving in the real number field. And control also takes values in real numbers, and the parameters also is real valued and so on and so on. So this is essentially a scalar second order system.

Then, we still want the control to be such that x1 tracks a desired trajectory, and we only give a tracking for the x1 and not x2, because subject to the matching condition, the desired trajectory of x2 is dictated precisely by the desired trajectory for x1. So, this is not something that is left to choice. So that is how it works.

So the, we should write the objective, in mathematical terms, it comes out to limit as t goes to infinity x1 minus r goes to 0, and limit as t goes to infinity x2 minus r dot 0. So, like we already mentioned, this double integrator system is in fact, a standard model for several mechanical systems on Euclidean spaces.

So that is, so there is already a lot of motivation for looking at systems like this. So subsequently, when we look at in a robotic systems and spacecraft dynamics, where the spacecraft dynamics is not exactly like this, but it is like a nonlinear integrator, nonlinear double integrator, and the robot dynamics looks pretty much like this. Except that these x1's and x2's start to belong to vector spaces, so R3 and R4 and Rk and so on and so forth. And, not just scalar values, that is pretty much the only thing that changes when you start looking at other real mechanical systems.

So, this is where we stopped last time we introduced this problem, and today we are going to start, designing the adaptive control for this system. So this is where we begin, we are on lecture number sorry apologize 6.5. We are on, lecture number 6.5, so is the fifth lecture of the sixth week, and this is the error model, just like before, so all the steps will start this rather similar to you, just like before you design an error model. In this case, there are two errors because, of course, there are two states x1 and x2. So, I construct an error corresponding to each state, and its objective value.

So objective for the x1 is r, objective for x2 is r dot. So, again, if you start thinking in terms of mechanical systems perspective, x1 is sort of the position variable or position state, and x2 is the velocity state, simple.

So then, as always, we again, once we have an error model, we want to cancel once we have an error definition, we want to construct the dynamics of this error. So that is what we do. And you carefully take a derivative e1 dot is simply x1 dot minus r dot and x1 dot is x2. Therefore, e1 dot is exactly equal to this step, which is e2, that is what we write here. So, this this happens, this equation 3.11 looks exactly like the original system only because of the matching condition, if this if r and r dot were not here, instead, I had two r dot r dot by k, then this will not happen.

And we already discussed this at some length last time that there is an infeasible trajectory. Trajectory has to satisfy some matching condition it has to be consistent with the dynamics. So I cannot make a system follow trajectory, which is not consistent with the dynamics. And the dynamics dictates that x1 dot has to be x2. So that is it. So that is what we have e1 dot turns out to be e2.

And then e2 dot is, just x2 dot minus r double dot, so the second derivative of r and so I just write x2 dot here and then minus r double dot and just keep moving on from here, so this now becomes the dynamics that we are going to work with this becomes our error dynamics. So, this is what we call. So, this is what we call the error dynamics, what we call the error dynamics.

Now, how do we do the control design for the known parameter case? So, what we do here is that we as usual try to find a target system, so, what is the target system that we choose in this case, we choose the target system as this e1 dot is e2, I cannot change that because there is no control here. So even in the target system e1 dot has to remain e2, but now in the e2 dot there is some control. So I can more or less dictate whatever structure I want for e2 dot.

So e2 dot is minus k2 e2 minus k1 e1. Why do we choose this? This is a system that we have analyzed even before using the Barbalat's Lemma and in strict Lyapunov construction, and so on and so forth. So we know and, and just by computing the eigenvalues of the system, also, it is a linear system, I can very easily conclude that this system is in fact, globally, exponentially stable at e1, e2 equal to 0, 0 is a globally exponentially stable system at the origin and that is why we choose this as a target system.

So there are two requirements, right? First, it has to have nice properties for the states, which it does it in fact, is globally exponentially stable. And second it has to be consistent with the dynamics which it is e1 dot is continues to remain e2, because I cannot change it. And here I can change the right hand side to almost anything I want. So I choose my stable right hand side, I stabilizing right hand side.

So, now in order to get from here to here, what is the control I choose? It is very simple, I just obtain it by subtracting the two. And so the control I choose to go from here to here is just canceling these two terms. That is this guy and that guy and then introducing these terms. That is this, this is just two pieces, cancel these two and then introduce these two and that is it.

Since, I have assumed already that I know the parameter value theta star, so I can in fact implement this controller in 3.14. This is implementable and I am good. So what is the next thing I do? I have to do a Lyapunov analysis. So, I have to choose Lyapunov candidate.

So what do I do, I choose my Lyapunov candidate as half k1 e1 squared plus half e2 squared. Now, we have already seen this kind of Lyapunov candidate before for the system, this is what we used for the Barbalat's Lemma type analysis. And we know by taking the derivative, so I am not going to take the derivative here, because we have done this couple of times.

We, we took this, then we take the derivative, and you plug in for the dynamics from here. So that is the directional derivative of V along this dynamics 3.13. You get V dot is minus k2 e2 squared. So, you do not get any term in the e1, you do not get any term in the e1.

So, once we understand that, we know that V dot is only negative semi definite, but we already have done this kind of we have already sort of dealt with this spring mass damper problem with exactly this Lyapunov candidate and this negative semi definite V dot, if you do not remember this, I strongly urge you to revise. I strongly urge you to go and revise our lecture on example of how to use the Barbalat's law.

So that is what it says V dot is only negative semi definite. So, we can utmost claim uniform stability from the standard, Lyapunov theorems, but then we can use signal Chasing or La Salles's invariance, either one of them. Of course signal chasing with Barbalat's Lemma or La Salles's invariance to show that both errors converge to 0.

On top of uniform stability, we can also obtain convergence by either using La Salles's invariance principle or by using Signal Chasing and Barbalat's. So, we have done this. So, again, for those who do not remember, I strongly strongly urge you to go back and revise this material.

So, of course, there is some nice terminology here which gets used in nonlinear control theory often these kinds of Lyapunov functions where the system is known to be asymptotically stable. So spring mass damper is in fact, asymptotically stable, and it is well known. But the Lyapunov function that we choose to analyze the system is only negative semi definite V dots is called a Non-strict Lyapunov function.

So, we know that the spring mass damper is asymptotically stable, it is in fact exponentially stable. We know that we can even obtain this result by simply computing the eigenvalues for this linear system. However, the Lyapunov candidate that we have chosen only yields the negative semi definite V dot, which does not allow us to continue from the Lyapunov theorems that system is asymptotically stable are called non strict Lyapnuov functions.

And in case of non strict Lyapnuov functions, as has been mentioned, you have to use signal chasing plus Barbalat's Lemma, or La Salles's invariance, in the special case of time invariant systems to conclude asymptotic stability. It is a sort of weakness, it is a sort of weakness of I would say, our analysis method. The system is known to be, known to have the nice property of exponential stability or asymptotic stability. However, because of our say, not so good choice of Lyapunov candidate, we cannot conclude that from Lyapunov theory.

So there is a lot of research in nonlinear literature, on non linear control literature on identifying good, strict Lyapunov functions for different kinds of nonlinear systems. So that is one of the problems that non linear control theorists do focus on.

So, now that we understand that, so let us, let us look at this. So, now we want to do the unknown case. So, what is it we use the certainty equivalence principle. As is mentioned here, CE is the certainty equivalence principle. What does it say? Keep the same control structure and replace the unknown parameter with its estimate. That is exactly what you have done. All other terms are the same, the only changes in this where we had theta star earlier, and now we have a theta hat. And of course, we will prescribe an update law for this subsequently.

Now, if I substitute this altered control here, because it is not theta star, the term with theta does not exactly cancel out, but you are left with a theta tilde so earlier, this was not there, earlier, we only had this much in the known case there was only this much. But because now, we are assuming that the parameter is not known, we replace the true value by its estimate, and therefore, you will get a theta tilde term. And this is what we use to in fact, compute the update law for theta tilde.

How do we do that, we take the Lyapunov candidate that we already had. Remember, this is what we always do, even in the previous scalar first order system case, that is what we did. There, we have only one term half e1, half e square. So here we had two terms. So we take the same two terms, and then add to it a quadratic in theta tilde. And as usual, this gamma is positive and its called the adaptation gain. Again, just like the first order case.

So take the same Lyapunov candidate as the known case, and add to it a quadratic term in the parameter estimation error. And, now if I take the derivative carefully, and substitute, I get this first term, which I would have gotten if there was no parameter error.

And then I get a second term due to the parameters. Not too difficult. I mean, if you are not convinced, you can just go here and do this. And then can just go ahead and do this sort of computation quickly. So V dot will be k1, e1, e1 dot plus e2 dot minus theta tilde, theta hat dot over gamma. We have used the fact that theta tilde dot equals minus theta hat dot because theta star is constant.

Now, if we substitute, there, I have k1, ei, e2, plus e2 e2 dot, sorry, plus e2 e2 dot plus e2 and e2 dot e minus, e minus k1 e1 minus k2 e2 minus theta tilde f minus theta tilde theta cap dot by gamma. So I have substituted for e1 dot and e2 dot, but theta cap dot is still not been decided. So that is being retained as it is. Now, it is not difficult to see that k1 e1 e2 and this k1 e1 e2 cancel out. And, what am I left with? I am left with minus k2 e2 squared plus if I write minus theta tilde, e2 times theta hat divided by gamma. Let us see the sign has turned out to be correct.

I think this should have been if I am not wrong, this should have been plus, that is what the error is, this is a plus. And this will be a minus e2 f. This is all correct. And so that is how you that is what you get. And this is what you get. And this is exactly the same. Now, what do I do? I do the best I can.

As always, I do the best I can, I cannot make this term negative definite, because that would require me to introduce a minus theta tilde here. And I cannot introduce a minus theta tilde. I hope this is very clear to you, theta tilde is unknown, because theta tilde contains theta star, which is the true parameter value, which is unknown to us. Therefore, we cannot implement a minus theta tilde. So there is no way of making this negative definite.

So they do the next best, I just try to remove this term, because it is not a definite term. And in order to remove this term, I choose an appropriate theta hat dot to cancel this.

That is what it is, this is the appropriate choice. Once we make the appropriate choice, I will get V dot is minus k2 e2 square. And this is negative semi definite. And notice, again, notice the pattern.

The, although the V is different, the V now has an added 1 over 2 gamma theta tilde square, in the adaptive case, my V dot expression which is minus k2 e2 square, turns out to be the exact same expression that I get in the non adaptive case also, this is always the case, because this is how because of how we have chosen the Lyapunov candidate for the unknown case, this always ends up happening. That V dot will have the same expression as the known case, even though V is different.

So, now, in the known case also in the second case, it was semi definite, V dot was semi definite negative, and here also it is negative semi definite. So, we have no choice but to apply. Well, of course, we can claim uniform stability at best and using the Lyapunov theorems, and now we apply signal chasing. And we do the same steps. So, I am not going to elaborate too much on these steps, because we have already seen it do the same steps.

First is V is lower bounded and non increasing because V is basically one half k1 e1 square plus of e2 square plus 1 over twice gamma, theta tilde square. So, obviously, this was greater than 0 radially unbounded in fact, therefore, it is lower bounded at 0 and it is non increasing because of this. So, we have a V infinity existence and is finite.

Now, because V is finite, because of the fact that V is non increasing, we have that all these e1 e2 and theta tilde are bounded functions, L infinity functions. Then just like in the previous case, we can integrate both sides here. And we can actually obtain that e2 is an L2 signal, because we have V dot is integral from 0 to infinity again same step as in the case of the first order system.

And further e2 dot is also in L infinity, because e2 dot is nothing but this guy and everything in the right hand side e1 e2 and theta tilde are bounded. So, if I assume something on the boundedness of f for bounded arguments again just like the scalar case, then e2 dot is also bounded.

Now, e2 dot is bounded and e2 is L infinity and L2 and by the corollary of the Barbalat's. So, from the corollary of the Barbalat's Lemma, we can claim immediately that e2 goes to 0. So, we get again in the context of mechanical systems, a velocity. So, you obtain a velocity tracking. So we also obtain that, of course, e2 is the same as e1 dot, therefore we obtain that the derivative of e1 also equals to 0, essentially velocity tracking happens. So, difficult velocity matching in robotic scenario.

Now, we do this again do similar steps as before, now that we have proved e2 theta goes to 0, we will try to prove that e2 dot also wants to see. So, this is again standard steps, we set as the sequence of steps, the sequence of steps that again identical in the first order case also we did the same, we proved that e goes to 0. So, that we therefore, after that we tried to go on to get e dot also goes to 0, which is we eventually did. So, that is what we do here too. So, now, we know that e2 dot is integrable, because e2 infinity is 0, so e2 dot is integrable and since e2 goes to 0 as t goes to infinity, and with appropriate assumptions again on f you can also do that the second derivative of e2 is also bounded, because the derivative of e2 dot is bounded it means e2 dot is uniformly continuous.

So, by the original Barbalat's Lemma e2 dot is integrable uniformly continuous signal, therefore, it goes to 0 as t goes to infinity. As, before we have proved that e2 not only e2, e2 dot also goes to 0.

However, in the first order case, e2 dot contain only e and theta tilde. Now, the e2 dot contains e2 e1 and theta tilde. So, if we know that e2 dot goes to 0 and we know e2 goes to 0 we are still left with the fact that the sum goes to 0. All we can say is that the sum goes to 0.

That is what we write here. That minus k1 e1 plus theta tilde f goes to 0. But unfortunately, this does not guarantee that the position error is going to 0. Look at this other nonlinear function I have the position error sure, but then I also have theta tilde times f added up to it.

So, the summation going to 0 does not really tell me anything about the error e1 going to 0. So, therefore, in this case, I cannot even guarantee that the position tracking happens alright and this is what is called the detectability obstacle in adaptive control. And this happens, because we started for the with a Non-strict Lyapunov function for the original or the known system.

If you start with a Non-strict Lyapunov function for the known case, then you will end up with a detectability obstacle and what it means, what does it mean the detectability obstacle, it means that, you are you may not be able to prove convergence or tracking of all the states, you will not be able to prove tracking of all the states. And that is rather disappointing as an adaptive control theorist. I said we do not care about parameters and Lyapunov function.

But we definitely want the system to track the trajectory. As of now, we have only been able to pull nice boundedness and stability and things like that. And then the velocity matches with the desired velocity. We have not been able to say anything about the position. And this is called the detectability obstacle. Why does it happen? We know that it happens exactly because I choose a non-strict Lyapunov function for the original known system. So that is, how do we fix it? Choose a strict Lyapunov function obviously.

So, one of the other alternate solutions was sort of proposed by Romeo Ortega in the 90s is still a relatively active researcher so he is retired from CNRS in Paris. But, but he still continues to publish articles. But one of the interesting things he proposed was a Lyapunov of like function, which helps us avoid the detectability obstacle. So what is it? So if you consider the spring mass damper system, I mean, let us not look at the original system we had, but a simple string mass damper system.

And suppose that we want to x1 and x2, we want to prove that x1 x2 both goes to 0, we want to want to show x1 x2 converges to 0 as t goes to infinity. So what did he say? He said that if you choose a Lyapunov like function, now, because this is not Lyapunov function anymore, I hope that is evident. It is only positive semi definite.

Why is it only positive semi definite? V minus alpha k comma k is identically 0 for all k, this is k comma minus alpha is exactly 0 for all k. So there are values of states outside the origin where V becomes 0. That is not allowed for positive definiteness. And so therefore, it is only positive semi definite, it is not even a Lyapunov candidate, Lyapunov candidate requires that you have a C1 function which is positive definite, at least. So you do not even have Lyapunov function we call it. We call functions such as these Lyapunov like functions.

And, what was his claim? His claim was that I can use or we can use a function like this in order to prove that both states go to 0. So, you did not even care about coming up with a strict Lyapunov function. In fact, come up with a function is not even a Lyapunov function you went further down, in fact, in some sense.

But he could still show that both states x1 and x2 will converge to 0 as t goes to infinity using this kind of a choice of function V. So, how does he do that is what we will see in the upcoming session.

So, what did we sort of see today is the adaptive control for a second order system. And by the end of the session, today, we could only arrive at the detectability of state. And, what is it? The detectability obstacle is that because of a choice of a non-strict Lyapunov function for the original known parameter case, we could not prove that our parameters, or even the states, the position state in this case converges to 0. So it converges to the desired trajectory. And this is very bad. And we are definitely not happy with the result of that kind.

We just do not want this bounded trajectories and position states not matching the desired positions and that is definitely something that we cannot live with. So, we started to see this Ortega construction, which hopefully will help us alleviate this issue. And that is what we will see in this upcoming session. So, this is where we stop now. And, I will see you again soon.