Nonlinear Control Design

Prof. Srikant Sukumar

Systems and Control Engineering

Indian Institute of Technology Bombay

Week 7 : Lecture 34 : Backstepping method for control design: Part 1

Everybody, welcome to another class on Non-linear Control, yeah, SC602. So I believe we have already looked at more or less the entire analysis, entire breadth of whatever is the analysis that we intend to look at. Of course there are more courses on pure analysis like I said. Analysis itself is a very big area but I personally don't prefer to go into and spend so much time in the analysis that we are not doing any design. So we have already taken one step in the form of control Lyapunov functions to get into the design, okay. So now what we want to do is we want to continue getting into the design and today we will sort of look at the first formal design method, okay.

So CLF was of course a tool which is used for design, yeah, but that's not a formal method for design itself, okay. So starting today, so pretty much as expected, you know, at the end of the, you know, midterm and then beginning, you know, just after the midterm we typically target, we start with the design aspects of things, okay. So that's the idea, alright. So we are today going to start with backstepping and I am going to write a little bit today.

The note that I am using is by Daniel Lieberzon on and it's, I believe it's his free version of his book, Nonlinear and Adaptive Control, okay. So his book on Nonlinear and Adaptive Control, this is what we are using as a reference right now. So there is an interesting history, so I just want to put in a little bit of the history, historical context here. So the idea of backstepping was sort of seen first in Russian literature. Interestingly a lot of controls related work has been in pretty much Russian literature, yeah, so it's not such a big surprise, A.

M. Lyapunov himself also Russian. So yeah, so a lot of work in control literature you will see is in Russian language, Russian literature. So this backstepping also is from 1978, Melaks, Russian literature and then it was sort of extended, I mean in the US, which is I mean slightly later by these folks. And finally came into the book of KKK, yeah, I think this is, I've already mentioned this book. Yeah, basically these are a bunch of guys who understood Russian I'm assuming.

Yeah, so this is Kanelakopoulos, Krstic, Kokoto-Vish. This is sort of my standard text reference for adaptive control, very difficult to say, so if you can say it really fast. So I usually just call it the KKK book. So this is one of our standard texts for adaptive control and these are the guys who introduced the terminology, yeah, they are the ones who introduced the terminology backstepping, yeah. So in research sometimes more than the folks who discover it, folks who name the method are way more popular, okay.

So remember this, coming up with good names for the method is also important, okay. So somehow backstepping as the name stuck and you will see why. The idea of backstepping is primarily a way of constructing these control Lyapunov functions, okay. So basically I would say that backstepping and this I want you to remember, a way to design CLFs, okay. So you already know the power of CLFs, right.

If you have a CLF you know that there exists a controller, you know, I mean CLF and some small controls, some nice properties, you know that there exists a smooth, almost smooth controller and not just the existence of an almost smooth controller, you also have a formula, yeah, you have the Sonntags universal formula, okay, which essentially gives you this very nice way of doing control design, okay. So very simple, very standard. So backstepping is a means of constructing a CLF and this sort of helps a lot of people because a lot of folks come back and ask me how to construct Lyapunov functions. So this is one of the simplest methods, okay. This is one of the simplest methods where you will see how step by step you can construct a Lyapunov function.

Of course it has its own limitations, there is no general solution to anything, yeah. You will also see that, okay. But where we will start is with the notion of integrator backstepping. Why folks like Christik, one of the authors of this book picked up this backstepping notions is because it's very powerful as a tool, not just in control, nonlinear control but also in adaptive control, okay. A lot of adaptive control, modern adaptive control results are based on backstepping, okay.

So this is actually a very powerful tool. If you can master this well, you will be able to do a lot of nonlinear control design for a lot of different systems, okay. And we will try to see, you know, a couple of examples and then of course, you know, more and more to follow in homeworks and things like that, alright. Okay. So what is integrator backstepping? I start with the system, x dot is fx plus gx u where I say x is in Rm and u is in Rm.

So basically gx is typically of the form g1x all the way to gmx, right. So this is just a way of writing fx plus summation of gi ui, yeah, the way we have been writing control affine systems, okay. So we always now go along with control affine systems. Yeah, we are not trying to, you know, have some kind of nonlinear dependence on control because it is rather unusual, okay. In most cases, you can always figure out a linear dependence with the control, okay.

Alright. So now, as always we, you know, we assume a few things. One is that there exists a control Lyapunov function V0 x for above. Yeah, you might say that, oh, okay, you already made life too easy, that you already are assuming a CLF, okay. We will see how, you know, we will have simple solutions there. There exists a smooth control law u equal to k0 of x which stabilizes above, okay.

So we are assuming, obviously it should be evident to you that if I already have a V0 then the k0 is obvious, right. So in fact, if you want to be, you know, more precise you can say you just have a almost smooth control law but I am making my life easy by assuming that I have a smooth control law, okay. So I have not just a V0 which is a CLF but I have some, I also have a smooth controller available to me for this system, okay. And of course we are assuming that all the nice properties that V0, F, G, k0 are smooth, okay. All these things of course give me a lot of nice new properties, okay.

So I make my life significantly easy, okay. Now k00 is equal to 0, k0 satisfies del V0 del x times Fx plus del V0 del x times G. So Gx is less than equal to minus Wx. So basically W is positive.

Yeah. So why I am specifying the third point, although this is the definition of, seems like the definition of the CLF itself, why I am specifying it is I am sort of saying that this, with this control this sort of negative definiteness is obtained, okay. Typically if you have a CLF it means that you have Lf V0 plus Lg V0 times control is anyway negative definite, right. But in this case I am saying with this particular control this is negative definite. So if you want to sort of think of it sequentially you can just use the first step and use the universal controller to obtain this k0. And then this is obvious, right.

So with V0 itself using the universal controller I can obtain a k0 and with that this step is this is anyway obvious because this is the definition, yeah. But because I said separately that there exists a control that is why I am actually writing it out, okay. There is nothing special about it. So these three things, anyway for safety it is always good to write out all the assumptions. Somebody might always say that you assume too much it is already obvious, that is okay.

The problem typically comes up if you do not assume whatever you need, okay. If you under assume that is a problem. If you assume more it is okay. You know you are still giving a result maybe a little bit more conservative but that is still fine, okay. But like I said you can always start with a V0 and then you can go on with the universal control formula to get a k0 and from there this is evident.

It does not need to be written out separately. So only this is required, okay. All right, great. Now that we have these assumptions so we already know that this system that you have is already you know stabilized, okay. It is in fact asymptotically stable, right.

The way we understand asymptotic stability because W is positive definite, sorry because V0 is positive definite, right. That is the requirement of a CLF. So V0 is already positive definite and its partial and its directional derivative is negative definite. So we have asymptotic stability by Lyapunov theorem, all right.

So great. Now the question arises what happens if I add an integrator, okay. So what do I mean by adding an integrator? I am looking at this new system now which is x dot is f of x plus g of x xi and xi dot is now the control, okay. It looks like the same dynamics, right. The first system looks very similar but the control is now being replaced by a state and then the derivative of that state is actually the control, okay. This structure is very common for most aeromechanical systems, okay.

Because just think double integrator. All double integrators have this sort of a structure x1 dot is x2, x2 dot is u, okay. So already very close to a mechanical system. So spring mass damper systems already very close to this. Then even if you have nonlinear damper like you know pendulum system very similar because your theta dot is omega and then you have you know and then you have the omega dot has minus sign theta and so on and so forth.

So again you have a similar structure, okay. So most aeromechanical systems have this kind of a structure, okay. And therefore there is great value in studying this structure, okay. So just a second. Yes, tell me what do you think? No I mean all I did was you had this system, right.

So all I said was I don't have the control here anymore but I have the control in the next stage, okay. So that is there is a state here instead of the control and then the derivative of the state is this. Like I said it's not like we get this from anything. This is actually how backstepping works by working over layers of integrators and the motivation is aeromechanical systems. Most aeromechanical systems will have this kind of an integrator structure, okay.

We will see some examples. It's not difficult to see, yeah, that you have this kind of a structure. So the idea is if you have some nice result for this top system you can extend that nice result to this entire system. That's the whole idea. Basically what we will do is we will construct a CLF for this system, okay. We already have a CLF for this and we use this to construct a CLF for this entire system.

That's the simple plan, all right. So what is it? I will also write the aim. Construct CLF for above using V0 x and K0 x, okay, all right. So I have been given a V0 and a K0 for the original system and I want to construct a CLF for this new system using that. Now the simple logic here, I earlier had this if I could ensure that this quantity xi was K0 x, right. Then I have my original system which is already asymptotically stable.

I am in good shape, okay. So basically the logic is if xi is identically equal to K0 x then done, right, because this is actually a stable system now so x actually converges to 0, great, okay. And I already know not just that x converges to 0, I also know that K0 of 0 is 0. So basically if x goes to 0 xi also goes to 0 because if xi is K0 x and x goes to 0 then K0 of x also goes to 0, right. So I have obtained my stabilization like I want, okay.

Now the problem is xi is a state of a system. It is not a control that I can specify. I cannot actually ensure that xi is exactly equal to K0 x, right. I cannot just say that xi is I will give you a function of the state and my new state will just follow this function of state. That is impossible, okay.

So we do the next best thing. What do we do? We create an error, okay. By the way notice that just for us to be able to write this xi has to be the same dimension as the control, okay. I hope this is obvious, right, because I wrote xi dot is u. So if the xi the new state has to be same dimension as the control.

If not cannot use backstepping, okay. This is a restriction of backstepping, okay. This dimensionality has to be maintained. Again for aero mechanical systems works out well, yeah. Usually there is position and velocity as states and the velocity derivative is the control. So position and velocity is the same number of dimensions usually, right, because if you are in three dimension, three dimensional position, three dimensional velocities, two dimension, two dimensional position, two dimensional velocities.

So works very well for aero mechanical systems, yeah. For electrical biological systems may not be so easy, yeah, because here the physical degrees of freedom restricts how many states you will have. Yeah, for electrical systems and biological systems that may not be the case, right. There is no notion of, I mean there may be an equivalent notion but there is no obvious notion of degrees of freedom, okay, alright. So we cannot make sure that xi is actually equal to k0 x but what do we do? We try to drive this state to 0, okay.

So try that this happens and how do I try that? I construct a new V of x and xi which is V0 x plus half norm of xi minus k0 x squared, okay, alright. Basically see what do I mean, I am basically using the logic that folks in control just want to drive things to 0, yeah. So I am trying to drive something. Right now my original requirement was that I drive x and xi to 0, okay, but I knew from my previous knowledge that if xi is actually equal to k0 x then I drive x to 0 and in fact if x goes to 0 xi also goes to 0. Since this is not possible I try to make the difference go to 0, yeah.

I try to make this error go to 0, yeah. This is not like tracking. Tracking is very different from this. Don't think of this as tracking. In tracking I would try to find the difference with a function of time, okay. For example when I am looking at robot tracking a path, the path is specified as a function of time, okay, or a drone trying to fly a path or a trajectory.

Trajectory is specified as a function of time, okay. This is very different. Here I am trying to make the error with a state and another function of state to 0, okay. This is only done in backstepping and nowhere else, okay. This is an unusual thing actually, yeah. We will get to situations and a lot of geometers don't like this idea, yeah, because if you think about again the first state as position states, second state as velocity states, okay.

So backstepping requires a comparison between the velocity state and the position state, right. You are somehow subtracting or adding velocity and some function of position. So to folks in geometry and topology and stuff they freak out because they say how can you add velocity and position? What does it even mean? Okay. So it's a pretty valid question, right. If I ask you to add position and velocity, say this function was nothing but minus of the position, okay, then you are essentially adding position and velocity.

Then the question is what space are you even working in, right. So it's very difficult to wrap, for them to wrap their heads around it, yeah. For us who are basically more like applied nonlinear control folks, for us it's just a tool, okay. This variable, especially when you are working in Euclidean space, everything is Euclidean, right.

So position, Rn, Rm and so on. So I can freely add them, no problem. It's just another variable in Euclidean space. Problems happen when you are working with something that is like a not a linear space, not Euclidean, so something like a manifold. Then the position and velocity are not in the same space anymore, okay.

Position is typically on the manifold. For those who have seen anything in geometry, they will know position is in manifold and velocity is in tangent manifold, okay, or the tangent bundle or tangent space, yeah. So the tangent velocities are actually linear space, positions are in a manifold, yeah. It's very interesting. This is why geometry is in general interesting and you are looking at, so this is all natural when you are looking at rotations and so on.

Rotation is on a manifold. It is like just like you see angles are on a manifold. They keep repeating. Say they are actually sitting on a circle, right. Not unlike what we think, they are not on the straight line. Angle is not from minus infinity to infinity, right, because you just repeat, right.

If you think of it as minus infinity to infinity, you are just cheating yourself because you are actually just counting the repetitions also, okay. So they are actually on a circle. Similarly, if you go to three-dimensional rotations, they are on the SO3, another manifold, all right. So for geometry is difficult to digest, but you can make sense of it.

Here in Euclidean spaces, for us it's very easy to make sense. You are just adding two variables in Euclidean space. It is just giving us some other variable. We don't care if it is making sense or not, okay. It is just for the purpose of analysis, all right. Eventually, after all this monologue, the point is that this I claim is a CLF.

This I claim above V is a CLF for X psi system, okay. So this is the claim, all right, okay. Thank you.