Nonlinear Control Design

Prof. Srikant Sukumar

Systems and Control Engineering

Indian Institute of Technology Bombay

Week 9 : Lecture 52 : Feedback Linearization: Part 4

What is a good plan is I know I have given this as a homework but I am going to just do this now. We will just do this here in class. We can assign something else for the homework. This is basically a DC motor model and what I had asked of in the homework was to basically verify the two lemmas point 1 and point 2. So what I am going to do is basically try to just let us see if we can do this on our own and if we remember all the ingredients of this lemmas. So that is basically what I wanted you to verify.

I think that is visible. Yeah. So you see that there is some nonlinearity here and here. Let us not worry about what the parameters are and so on, what theta is, what AB is and so on.

That is not our concern right now. We just want to verify these lemmas. Okay. And we are also given an output with respect to which we want to work. Now the first thing is what was lemma 0.

2? Sorry 0.1. We are just verifying that all this if you are given that LGH equal to LG LFH all the way to LG LFR minus 2 H is 0 then you get that L, yeah I am just going to copy this I guess. Then you are, you can basically claim that all the L add FKs are also 0. Okay.

Yeah. So you can claim that L add FKGH is 0 for K equal to I guess 0 to R minus 2. Okay. Because I am just specialized it to relative degree R.

Okay. So the first thing we want to do obviously is to compute these. Yeah. Let us start computing. So first things first what is F? What is the drift vector field? What is the second term? I think this is also a controller.

Wow. Wait a second. Bx 2 plus ua that is some K. Yeah. Looks like it is just some constant.

Okay. So this is K. Yeah. Good handwriting. Alright.

Alright. Fine. What is F? Let us start. Minus Ax1. Okay. Second component? Yeah.

The whole thing right. Third component? Excellent. Everything without the control. And G is? Excellent.

Yeah. Great. Okay. What is h of x? Is x3. Yeah. Of course one simple way to do your find relative degree is to start taking derivatives directly and find relative degree. But unfortunately I have asked you to verify the lemmas.

Yeah. Or ask myself I guess in this case to verify the lemmas. So we will actually have to compute all this. Alright. So if you see that the first term here is also Lgh.

Right. Because if I put K equal to 0 this is just Lgh. Right. So let us just compute Lgh first. What is Lgh? First I take partial of h with respect to x. What is it? partial of h with respect to x.

0 0 1. Thank you. And G is just 1 0 0. So obviously 0. Done. First done.

Let us do Lgh Lfh. Did I get this correct? Yeah. I think I got this correct. What is Lfh first? Lfh is again 0 0 1 multiplied by this mess. Minus Ax1 minus Bx2 plus K minus Cx1x3 theta x1x2.

Okay. What is this? Theta x1x2. Thank you. Alright. Now we do Lg of Lfh. Or Lg Lf I can put the bracket that is fine.

Yeah. So this is the distributive operation. Now if I want to do Lg of Lfh what do I have to do? I would take partial of this guy. Right. What is it with respect to x? Yes.

Go for it. Theta x2 theta x1 0. Theta x2 theta x1 0. Now and I multiply with G. Now did I get something? Thank god I got something.

Yeah. This is what? Theta x2 right. Okay. Done. What was relative degree? Relative degree is what now? R minus so this has to be 0 until R minus 2 right. So it was 0 until K equal to it was 0 until what? 1.

Relative degree is 2. How did I get 2? From this expression how did I get 2? I got that only Lgh is 0. So this term has to be Lgh. So R minus 2 equal to 0.

So R equal to 2. Great. Just okay. Just the last term has to be 0 right. All right. So then I have to of course do this also. I have to prove that I have to now compute L add f 0gh also.

But this is actually equal to what? Lgh which is already proven to be 0. So proving so lemma 0.

1 proved. Yeah. Too easy. Because I didn't have to compute any further derivatives at all.

Done. Okay. Great. What was lemma 0.2? It was the linear independence. All right. Linear independence of dh, dLfh and so on and so forth. Okay. In this case how far do I have to go? Only till R minus 1.

So what is R minus 1 in our case? 1. So I just have to go till Lfh. Lf1h whatever or Lfh. Yeah. So this is that is it.

So I just have to compute this much. dh and dLfh. What is h? Sorry. What is dh? You have already done this. It is 0 0 1.

What is dLfh? Did I get this right? Yeah. dLfh is, dh is correct. Lfh is this guy. What is the d of that? You already computed that also.

Yeah. Theta x2, theta x1 is 0. Okay. This is linearly independent.

Right. I mean okay. Let's be precise. Right. If you remember all we always talked about this linear independence for a particular point. We said at x0. In this case this is rank 2 for all x1 x2 not equal to 0 0.

Okay x3 can be anything. Notice x3 can be anything. It is irrelevant. x1 x2 both cannot be 0. Okay. And of course you can assume theta is not 0 otherwise stupid system.

Yeah. Alright. So both x1 x2 cannot be simultaneously 0. Okay.

x3 anything is allowed. Okay. Then this is rank 2. We are done. We have coordinates. Right. So what are our coordinates now? y1 is hx itself which is x3.

y2 is lf hx. Right. That is what we chose. Which is what? Theta x1 x2. Yeah.

That is it I think. You only get two coordinates. Right. Because I mean how many do we get until r minus 1. Right. We get until r minus 1.

r minus 1 is just 1 in our case. So we get r minus 1. Right. So we get r minus 1.

Right. So we get r minus 1. Right. So we get r minus 1. Right. So we get r minus 1. Right.

So we get h and lf h. These are the two coordinates we get. Now what is the third coordinate? I don't know. I don't know what I can choose. I just want to make the whole thing a diffeomorphism. Alright. So as of now what did I get as the Jacobian? I got it as 0 0 1 theta x2 theta x1 0.

Yeah. Now what should I choose as my third coordinate? Any guesses? Basically I need it to be linearly independent from y1 and y2. That is in the Jacobian it should contribute a third dimension. What do you think? I think something like minus x1 and x2 0.

This will work. Right. Because that makes these orthogonal. No. This makes it orthogonal. Right.

I mean this dot product of these is 0. Right. I just used that idea. So if this is the Jacobian then I am good because this will be rank 3. Because if you notice if you take the determinant e to x this is always ok no problem. Yeah. Always going to give me one independent row or column.

The determinant of this is what? Theta x1 square plus x2 square. Right. So if it is not at the origin it is ok. That is the best I can do. So I can take this. So what will be the y3 such that this is the Jacobian? Partial with respect to x1 is x1 partial with respect to x2 is x2.

So this is half x1 square plus half x2 square. Yeah. Alright. So I sort of back calculated this. Yeah. No point adding anything here.

I see this is the trick I am just paying a trick here. Right. I am just adding a state so that I get this diffeomorphism. No. That is all my aim is. So adding anything here is pointless because already have you know full rank in this.

No problem. No point adding anything here. How to make these two linearly independent? Make them orthogonal. The best way to make any two vectors linearly independent is make them orthogonal. This is the best way to do it. I just do x1 square by 2 and x2 square by 2.

Look at this very crazy looking coordinates I got. Right. Unusual. Started with very nice looking output. Yeah. But then I ended up with almost looks like my Lyapunov function.

Right. I got almost a Lyapunov candidate in the first two states. But whatever this is a fair set of new coordinates. Okay. Now what will happen? I mean if I actually write the dynamics.

Yeah. What is the minus? You guys always wait till the end to tell me.

Which one is minus? This one. This one. No. Thank you.

Alright. So this is the Lyapunov function. Excellent. I like that. Alright. Whatever. Right. Yeah. Funny system. I mean I wonder that can it be simpler? It could be but then what will happen is I could try something simpler.

What will happen? That do is it will restrict the x1 x2 in which this will work. Right now it will works for a very good class of x1 x2. Basically whenever both of them are non-zero it works. Right.

Which is what was the assumption here also. So this works whenever both of them are non-zero.

Sorry. Any of them is non-zero. Not both of them. Any of them is non-zero this works. Right. That is pretty good. Right. Because usually I always want to drive my systems to zero.

So it works all the way till the end. Then it does not work, it does not work. I mean you can see that I work almost till the end. Right. If I choose something simpler I think it will become restricted.

If I tried anything else. If I remove this for example if I remove this guy. If I made this zero. What would that do? Determinant will be theta x2 squared.

Right. So x2 will have to be non-zero for this to work. Right. So yeah. I mean you can think of other fun choices but yeah. Let's try to compute the dynamics in this new variables.

Let's see what I get. y1 dot is what? x3 dot. x3 dot was what? Theta x1 x2. Oh that was expected. y1 dot was supposed to be y2. Sorry that is the whole point of this whole exercise.

Yeah y1 dot is y2. Yeah. What is y2 dot? y2 dot is the derivative of theta x1 x2. So that is theta x1 dot which is ax1 plus u minus ax2. So that is theta x1 x2 dot which is ax1 plus u minus ax1 plus u times x2 plus theta x1 x2 dot which is minus bx2 plus k plus cx1 x3.

Right. Something funny. The only thing is my control appears here. Only thing is the control appears here. Again it is very difficult to actually write this in terms of the y1 y2 and y3 so I am not even going to try to attempt this. But the control appears here. That is the good thing. Yeah and then if I do y3 dot I get half x1 square half x2 square so the derivative is minus x1 x1 dot which is minus ax1 plus u plus x2 x2 dot which is x2 minus bx3 plus ax1 x3.

Say again if you see the control appears here also. So the control appears now in two equations. Could have been avoided again if I got rid of this guy and got rid of this guy. Control would not appear here. Then you would have control only one equation. May have made control design easier. So this now then choosing the third one depends on a lot of factors of course you're trying to make a diffeomorphism but there are infinitely many choices.

Then it is really on what's good for you. So I think I don't know. Did I try to.

Yeah I did something interesting here if you see. I chose this. Yeah. You see the previous page. z1 was x3. I used the notation z1 z2.

z1 was x3 of course z2 is theta x1 x2. These we have no choice. It is what it is. z3 I took as x2 minus k over b. Okay. Now why did I do that. Then because I got z1 dot is z2 z2 dot is whatever.

And z3 dot is this nice expression. I have no idea why I chose this. But if I take z3 as what I chose x2 minus k by b then what happens.

Look at this. What happens. If I choose z3 as x2 minus k over b. Then this guy the third row becomes what.

The derivative right. It is just the partial of this. So it will be 0 1 0. Because this k by b is some constant.

Right. This is also not a bad choice actually. You think about it. Right. Because it gave me a good extra column. Right. And then this is what it is. If one of them is any one of them is non-zero I am fine.

So actually this was a simpler choice. Right. Because this is just actually linear. Right. So linear transformation is always nice. Right. Starting from one coordinate you move to another coordinate in a linear way.

This is also a nice choice. Because it gave me a nice full you know. These two are already linearly independent.

Depends on nothing. Okay. And then this one all you need is. Well actually in this case also. No it is not that nice.

In this case also you must have x2 to be non-zero. I apologize. No it is not that nice by the way. Yeah. Sorry. You will need x2 to be non-zero.

Because if x2 is zero then this goes away. You could have something here but then this becomes the same. Yeah. Yeah. Yeah. So this will also need x2 non-zero. Yeah. Whereas this choice even though it is ugly looking.

So this is how you work with feedback linearization. Eventually all those complicated looking expressions you only use some part of it. In this case you only used r equal to 2. So those expressions are significantly less in number.

And the computations are exactly like this. You have to do this partial then multiply and so on and so forth. Okay. No, any questions? Then you will have full state feedback linearization which is what we will see next.

In that case it is the system is completely linearizable. Yeah. Physical meaning. Yeah. I mean it has physical meaning in the sense that it connects how your output and input are sort of you know. Again if you are looking in terms of whatever is your r your system has a subsystem which is an r dimensional transfer function. Basically your, you know the larger system with additional state but then in the middle there is sitting this r dimensional linear system. So it is almost like saying there is yeah.

I mean then there is these notions of I mean I mean I unfortunately will not have time to talk about those. But there is this notion of differential flatness also. Some of you have seen this. Basically it says that you take a certain number of flat outputs and all the system states and the inputs can be represented as these flat outputs are the derivatives.

So again this is also similar notion. Right. There you are it is a property of the input output system. Yeah. It is somehow making some part of the system linear.

So the behavior is will be linear. Yeah. Whatever all the robot dynamics cases I mean it is fully feedback linearizable. Right. So it actually behaves like a double integral. It behaves like a linear system. Particularly easy to control particularly easy to work with.

Other than that no I don't know of any other physical relevance. In controls folks attach physical relevance to sensors and actuators. So if there is a linear connection between the sensor and actuator we really love it.

We can do so much. Yeah. We can even you know most importantly why because we can get performance. We can talk performance transients. Nonlinear systems. Yeah now there is so much literature on transients but linear systems so mature right.

How much overshoot, how much you know damping, how will the oscillations look. These are also like critical values. Critical. Yeah. So that is the physical relevance for us.

All right. Yeah. Fundamentally there are robustness issues with feedback linearization. It is not Lyapunov based.

As you can see you are canceling things. You keep trying to cancel things. Yeah. Even this dynamics we saw in the end. If you see you got this structure. It is a rather complicated looking structure here. You got the control here and you got the control here. Notice you had u x 2 and minus u x 1. You see this orthogonality sort of in the control.

You remember in a Lyapunov function things you just take it as you know if you take x 1 square plus x 2 square it sort of cancels the 2 in some sense. Okay. But you have u x 2 theta and minus u x 1. Of course you can have a theta. It does not matter.

You will see that you are not asking for x 1 and x 2 individually to be non-zero but one of them to be non-zero at least. So if one of them is non-zero. Say x 2 is non-zero. Yeah.

And this is zero. Then you cannot do anything with this state anyway. But this state you can make this whole thing linear by canceling all the non-linearities. Right. So because it is a relative degree 2 system you can only get a 2 dimensional linear system.

Similarly if x 1 is non-zero then this system can be made linear. This still remains whatever it is.

Okay. So you again get a 2 dimensional system which is linear. That is the whole idea. Alright. Okay. Any further questions? Alright. We will stop here. Thank you.