Course Name: INTELLIGENT FEEDBACK AND CONTROL

Professor Name: Leena Vachhani

Department Name: Multi-disciplinary primarily for Mechanical, Electrical, Aerospace and Chemical engineering streams

Institute Name: Indian Institute of Technology Bombay

Week - 02

Lecture - 09

Hi, in this video we will look into one more example where the PID control kind of a structure is working for a non-linear system. And this further explained why Jugaad with the help of PID controls are giving solutions. All right. So let's understand this is this example of controlling the hovering of an unmanned aerial vehicle, and I'll be using the term UAV referring to the unmanned aerial vehicle in this video. So let's understand the system first and then design controller and that will help you to see how we make sure that the system representation turns out to be in a simpler form appropriate for the PID controls to apply.

So the UAV position is given by the three-dimensional coordinates, Cartesian coordinates x, y, and z with respect to the earth frame. And the UAV orientations are given by psi,theta,phi in three dimensions again, which is yaw, pitch, and roll. The corresponding angular velocities are given by PQR. Now, my hovering objective here, hovering means one has to stay at a particular position, three-dimensional position and angular position for some time till the hovering command is given. So, control objective for the hovering now involves this particular Z to remain at a particular set point, or the Z represented in the earth frame given by a set point.

In order to get the simplified mathematical model, we will make some assumptions here. The assumption is that the body frame of the UAV is perfectly aligned with the global frame. Means when we are starting the quadcopter's frame of reference, which is sitting here, is perfectly aligned with my earth's frame or a global frame here. And all the initial states of the UAV are considered to be zero. For example, we have already achieved this particular set point.

Now the objective is to remain there. So my errors are all zero and therefore initial state is zero. Now UAV is controlled only by the virtue of the thrust given. Thrust means up and down thrust. This is what the simple command given and we would like to make sure that this set point is achieved or any change in the set point is being achieved here.

All right, fine. So now if I look into the dynamics of the system, it turns out that I can simplify by writing this particular equation where this RBE is the transformation from body frame to the earth frame because my thrust is being applied onto the body and you require a body frame to earth frame transformation. Now, since we are only looking into the Z parameter or the Z variable to be controlled, and it turns out that because we had considered the body frame alignment with the global frame, we can consider the dynamics associated with X and Y accelerations as 0. Now, this transformation is very well known in the robotics community and the C and S terms turns out to be just the simple form of representing cos and sin. Now, these above equations represent the UAV multirotor guidance model of MATLAB.

One can also refer for further understanding of this equation. My interest here is to get the dynamics for ZE, which is where we want to make sure that this ZE follows a particular set point. Now, the dynamics of the system is now given by MZE double dot equals ME MG minus F thrust. So now here I can write this particular equation and make sure that my dynamics turns out to be in this particular form. Now, this particular dynamics is nonlinear.

Why is this nonlinear? Because this particular g term gives me the biases here. We know that y is equal to mx plus c, where c is my offset is a nonlinear equation, though we say it's an equation of a line. It doesn't pass through origin, as long as it doesn't pass through origin, you will not be able to apply the superposition. And this violates the LTI assumption here.

So you can see here that F-thrust is the input to the system. Z is my output. M and G are parameters of the system. Now this mass M of the UAV scales the effect of this F-thrust. Because of understanding, we can see that this is dividing this.

It is the multiplicative factor here. So this F thrust is scaling the F thrust, the input U, whereas G is a constant bias. Therefore, this particular equation represents a nonlinear system. Now, how to make the dynamics linear? Because finally my objective is a simplified control design with the help of PID.

So, I will have to consider some kind of jugglery in order to represent this nonlinear form to a linear form. So one way of doing it is by considering a virtual control V, which is given by F thrust by M minus G, and now my Ze dynamics is linear because my Z double dot equals V. So what we are doing here is now we will try controlling the Z with the help of this virtual control. So my virtual control V is applied to this system. Now this becomes my system equation.

This particular transformation will also sit and at the same time inverse transformations will also sit in order to make sure my system represents like this. But at the same time, this particular system is a second order integrator. All right. So double integrator and design and the control of the double integrator can be very well handled with the help of derivative control. All right.

So with the help of this particular idea that my system is now a double integrator. Now, double integrator needs to be designed with the help of the PD controller because any PI controller will add more lag to the system, which will not be useful for us to design this controller. At the same time, a simple negative feedback control action should be just sufficient to satisfy this particular control objective of regulating. This becomes a regulation problem of making sure that this Z variable is following a set point here. So we have already said a suitable control scheme is a PD controller.

All right. So if we look at the step response of this double integrator, this is an open loop response, which is a growing amplitude with respect to time. Now, if I apply a step input, it's going to be a growing output. But whereas if I have a PD control, then the transfer function of the system with PD control turns out to be nothing but GH pi 1 plus GH, which is given by KP plus KDS because my controller is now the PD controller. And if I look into looking into any such PD control, this is helping me to reach to this particular setpoint giving me the step response, and the double integrators response to a step input.

Now, with PD control turns out to be achieving a set point, a settling time equal to the set point. Now, interestingly, what we did in this exercise, we would like to understand it in a wholesome manner. What we did is we simplified the model and then in the process of this simplification, we got that this particular system model behaves like a double integrator. Since it is a double integrator, we have no choice but to use PD control. And with the help of PD control, we are able to achieve the hovering mechanism.

And this hovering objective is achieved for a nonlinear system. But this simplified process gave me an opportunity to explore a simplified controller, which is nothing but a PD controller. Of course, we got some kind of an overshoot over here. But finally it is reaching to the steady state value. At the same time, we have to understand now this transformation and where is it sitting.

All right. So what we have is a plant which is now which is nothing but our UAV. The output of this plant is my Ze whereas the input to this plant is Fthrust. We have seen this, correct? Now we are applying the control over the transform variable V, so we should have a transformation.

Since this particular control is applied on a transformation, on this particular virtual variable V, this is the transformation which is turning out a Z set point to a V set point now. Now what we had was the equations, let me just simply copy these equations for us, which is V is equal to F thrust by M minus G. So we have this relationship V is equal to F thrust by M minus G, so we can find out what is my V set point here and apply this particular controller. Okay now we are doing this particular v set point, this v to f transformation of course here is a pd block sitting here and then there is a transformation which is converting v to f thrust Which means F thrust is the output.

So I should need the inverse transformation given by M times V plus G. So this transformation is taking M and G values to convert error in V to the thrust value. At the same time, since we are applying the control over with the help of this particular virtual variable V, what we have is we need the measurement of ZE, which needs to be pipelined to give the input to V. And this is possible when we say that, OK, here is V is equal to ZE double dot. So this particular transformation and this particular inverse transformation is needed in order to make sure that there is a normalization happening somewhere. And this also gives an idea that how to use some particular nonlinear transformation.

This turns out to be our nonlinear transformation or inverse transformation. And this is my nonlinear transformation in order to incorporate the PD controllers for the linear systems. And so this this exercise has given an idea that simplification of the nonlinear. First of all, we did simplification in terms of the multiple variable control to a single variable control. Multiple variables for X, Y, Z and angular rates.

But we were interested in only hovering command to be given. So that's where we simplified first that, OK, it's a single input, single output case. Now, with the help of single input and single output, again, the system is nonlinear. How do I do the linearization? Can I do it with the help of virtual signal way?

Or I can always consider an operating system to linearize the system. But in this case, it turns out finally the control is PD control, which is a very simple control methodology. And this gives me this answer in a jugaad way. Thank you.