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 - 01

Lecture - 06

Okay, so we saw one kind of PID tuning method which was ZN method. Now, we will look into certain other methods and those methods are also very popular. One of them is chien, hrones and reswick (CHR) method. Now, we also know that ZN method was used for getting the quickest response in terms of the amplitude decays, the quarter amplitude decay was the criteria. And let us see what is the criteria followed in CHR method now.

CHR method for the closed loop systems will give a better robustness as against the ZN method. For this particular CHR method is designed for quickest response without overshoot. There are two tuning rules given. One is the quickest response without overshoot and the other is quickest response with 20% overshoot. The way ZN had done, ZN had said, okay,

Let me give you one rule which is based on step response or the frequency response and they will always be making sure that you are getting the overshoot and that particular overshoot will die with the quarter amplitude decay. Whereas CHR said that okay let me give you the tuning rule based on no overshoot and 20% overshoot. At the same time, there are two different control objectives considered. One is tuning rule if you are considering the load disturbance rejection as the criteria, then there is another one set of tuning rules. And if you are considering set point response as the control criteria, then there is a separate tuning rule.

Now, if we compare between this no overshoot and 20% overshoot with respect to the tuning rules for set point response method, see that the proportional control tuning rule is same. So, for example, for no overshoot, you got the proportional gain as 0.3. Similarly, even for the load disturbance, you got the proportional gain as 0.3. And same for 20%. Fair enough.

But as soon as you introduce integral control, you can have better idea about what the load disturbance and set point response could be. So there comes the difference between what, I mean, you have to refer to this particular table. What do you want? You want load disturbance with no overshoot? Or you want set point response with 20% overshoot?

Refer to this table, and based on that, the tuning will change. This also gives a fairly good idea that okay, proportional control you have limitations. You will not be able to do a very good load disturbance rejection or a very good step point response. You will have a fairly okay idea about it. And with or without overshoot, you have the same kind of response that you will get.

But if you want to go ahead with PI or PID control, then you have to decide the control objectives. what is the priority for you, whether the load disturbance is a priority or the set point regulation is the criteria. Based on that, you choose. Now let's look into the comparison between CHR and ZN method. What we have is case of the ZN is also designed more or less for the load disturbance rejection.

And at the same time, if you look at CHR, no overshoot, 20% overshoot versus the ZN tuning rule. These are the tables kept one after the other for you to have a comparison here. So we look into this comparison. What is the observation here is that the control parameters for ZN tuning rule matches best with the 20% overshoot criteria. So we have this particular AK.

You see proportional gain is 0.7 over here. Proportional gain is 1 here. And TI by L is for PI control is 2.3, which is more close. And at the same time, you get something like similar PID tuning rules here. We already know that ZN tuning rule is having the criteria of quarter amplitude decay, which fits.

So more or less, CHR is taking care of that decay as well. That we can conclude from here. But what is the drawback with CHR methods is that, okay, this method will also suffer from the decay ratio being too large. So here, it was, if you consider between ZN tuning and 20% overshoot, this decays fast, which means that the closed-loop system will have poor damping and high sensitivity when we look into the CHR tuning rule. There comes further refinement for the tuning rules, and Cohen-Coon method is another one, which gives the tuning rules for the process transfer function for FOTD.

So we have the FOTD model here, which is having the parameters Kp which is gain, process gain, the time constant T and the delay L or the lag. In this case also, Cohen-coon also says that, OK, let's have the design criteria as the rejection of the load disturbances. Now we will have the we will have the dominant poles position being considered for the controller design, that will give the quarter amplitude decay ratio, similar to the objectives of the ZN method. But of course, this is applied on more rigor theoretical way and so on and so forth. If we consider similar to our methodology in ZN, the parameter A turns out to be KPL by T and tau turns out to be L by L plus T.

And they said, okay, now let's have the rule for P, PI, PD and PID. These rules are designed nothing but by applying a dominant pole which will give the quarter amplitude decay ratio. The idea behind ZN method and Cohen-Coon turns out to be same. Okay, so their objectives are same. But they are applied on, ZN tuning rule was designed for the integrator kind of a process, and then they made some work to say, okay, now if my system also has a time constant, then let's change L by L by T and so on and so forth.

Here, this has a little more theoretical background, and that fits well if we consider these parameters A and tau. All right, so now if we summarize, we have ZN tuning rule, which is simple and intuitive, and that's the reason it is more or less very widely used in order to consider the control parameter tuning because it requires a very little process knowledge. We can even design it with the help of the step response and the frequency response way. And we have also said that, OK, if we are considering a frequency response method, the experimental way of getting the ultimate gains and ultimate frequency is very, very easily done. And that's the reason we should be able to design the control tuning controller parameters here.

In both the ways, the process is characterized by two parameters that we get from the frequency response or the time response way. And now comes the drawbacks of ZN methods. Since it is simple at the same time, it requires very little process information. So it uses also process very, very less information. So in case you have a better understanding of the process, you have a better process model available, then Ziegler-Nichols doesn't give the opportunity to use that those information.

And that's the reason, though it is simple and intuitive, we may avoid using it if more process information is available and we may go for Cohen-Coon method way to consider the tuning. At the same time, if we go for ZN tuning rule, we also know that this particular design criteria is quarter amplitude decay If we consider this particular time based control objective, then it provides very low robustness. And that's where one has to be mindful of this when we are going for this tuning using the ZN method. As compared to ZN, CHR method gives slightly better robustness.

And Cohen-Cohen uses, since it uses three parameters and still uses quarter amplitude decay as a design criteria. So it provides better robustness. Again, better robustness and load disturbance rejection as compared to ZN method. But at the same time, we also know that ZN is very simple method, very intuitive method, so we can use at certain places. Especially when the process is completely black box to us.

Now, we see that ZN tuning rule, Cohen-Coon or CHR methods give us a very intuitive ways to design the controller. At the same time, we will have to do some kind of manual tuning even when these methods are giving you some approximate controller values. It is because finally, these these methods are experimentally obtained values. Fair enough. But at the same time, when we are actually performing, actually connecting the controller to the process, we will further be needing some kind of manual tuning.

So when we are doing the manual tuning, we can apply some simple rules here. What are those simple rules? So when we are designing the controller and we also have to give this controller methods to an unskilled person, we have to create the procedures for designing the controllers. So ZN method or these tuning rules give the basis to start with. So we start with and then we do the manual tuning for the fine tuning of the control parameters.

Now here comes what should we apply rules, some procedure we will have to give it to the unskilled person to design, to do the fine tuning as well. So what we can consider here is that if we are increasing the proportional gain, stability is affected. And how is that affected? When we increase the proportional gain, we know that the stability is going to decrease. We start with the tuning rule method.

The basis is my tuning rule method. Let's say we design PI controller. We start with designing it by tuning it with the ZN method. We got some values. Now, when you are doing the manual tuning for the fine tuning, we increase or decrease the proportional gain if we want stability to be decreased or increased respectively.

If we increase the, we decrease the integration time or increase the integration gain, then steady state error will decay rapidly. This is a complete, of course steady state error is going to be zero if we are having the integration controller already introduced, but the steady state, or settling time is going to be faster if I am having the integration time is decreasing. If I am decreasing the integration time, the stability is also affected. If I am decreasing the integration time, stability is also decreased. If I'm using the derivative controller and I'm increasing the derivative time, then stability is further affected and introducing the higher derivative time improves the stability because finally you are adding the phase lead to it, and your point on the Nyquist curve goes away from the ultimate point.

And that's where you get better stability margins with the introduction of the derivative controller. But yes, derivative time constants need to have some kind of an upper bound because otherwise your noises and other things gets picked up. These are the practical limitations of using the derivative controller. Fine, so these are certain ideas behind doing the manual tuning, when to increase and decrease. But there are further formal ways of doing the manual tuning.

And those formal methods are based on the tuning maps. Now, what are these tuning maps? Let's understand with this particular diagram. Let's consider a process transfer function, S plus one to the power minus eight, means that there are eight poles at minus one. Means it's a very complicated system, but its behavior turns out to be similar to some kind of a second order system and so on.

Now, For example, I have this ZN method for this particular system gives you K, means proportional gain as 1.13, integral time constant as this, and derivative time constant as this. It means I'm applying the PID control. So what is this particular chart gives you is that if I go from left to right, my integral time constant is decreased. So from 10 to 5 to 3. But proportional gain is kept constant.

But if I go from top to bottom, then my proportional gain is decreased from 1.5 to 1 to 0.5. So this way we are creating the matrix of observations. This way I'm changing the integral time constant or integral gain. Top to bottom, I am changing the proportionality. All right.

Now let's see the observations here and what could be the values that would be most suitable when I am changing these integral time constants or the gain values for PI controller and what should be my best case when I'm doing the manual tuning. So this kind of chart helps in a way. If I go from left to right, what we see here is that my settling time is reduced, right? It settles down to the origin. So these plots are for the error.

And this particular, you can see that it settles down faster. All right. But when I go from top to bottom, what is observed here is that the oscillations are reduced. Alright, so this way we keep going from top to bottom by decreasing the gain and seeing that where the oscillations are kind of better and reduced. And if we further go down, then my oscillations are further decreased, but at the same time, my settling time is also decreased.

But at some point of time, we would like to come to a conclusion where such that my best response is turning out to be in the middle of the matrix. And that's how we choose the manual tuned way of gain k equal to 1 and integral time constant as 5. OK. Something similar we can do it with Nyquist plots too. This was the earlier one was step response.

This one is of Nyquist tuning map with the help of Nyquist plots. Now, this Nyquist plots, if we see again, the same gain is increased. Gain is decreased from 1.5 to 1 to 0.5

when we go from top to bottom. And integral time constant is decreased from 10 to 5 to 3 when we go from right to left. And we see that of course we already know that the Nyquist plot if I am decreasing the proportional gain then what happens is this particular curve is moving towards radially moving towards the origin.

You can see here every when we go from top to bottom, this is going internally radially going towards the origin. All right. This one also going radially towards origin. Whereas integral time constant is decreased means the integral gain is increased. So your curve is moving towards the ultimate gain point.

So it is moving towards the top left side where our I axis is. Okay, so this is again gives you an idea what would be the best kind of response that you get. This is giving a fairly good stability margin because this is very much away from (-1,0) point, and that's the reason we can again fix this particular curve as the good point for the tuning. So both methods we can use for manual tuning, either the frequency response way and plot the Nyquist plots when we are doing the manual tuning, or we can use the step response way to consider it. So what we covered here is the PID tuning with the help of first some tuning rules and then do the empirical tuning by doing some kind of small fine tuning of the proportional gain, integral gain and the derivative gains and so on.

We also know that ZN tuning rules will give good response if L is less than 0.38T. You go back and put this value into the tuning methods, tuning table that we have in ZN method and see what kind of proportional gain values and integral gain and the time derivative gain values that you get. you will have a fair idea that this particular L, if this particular lag is more than 0.38T, far more than 0.38T, your gain values are turning out to be very different and proportional gain especially blows up. And that's where what we have to consider that the PID tuning works if your system is not a delay dominant system. If it's a delay dominant system, it's a very good case to consider that PID tuning is going to be very difficult.

So let's not try PID tuning, but resort to some modern techniques and so on. But if your system is such that your delay is fairly OK with respect to the time constant of the system and rather if it is comparable with respect to L and 0.38T. If your delay is comparable to

0.38T or lesser than that, then PID tuning is going to work. And that is the reason most 80 to 90 percent of the industrial controllers are still PID controllers. Thank you.