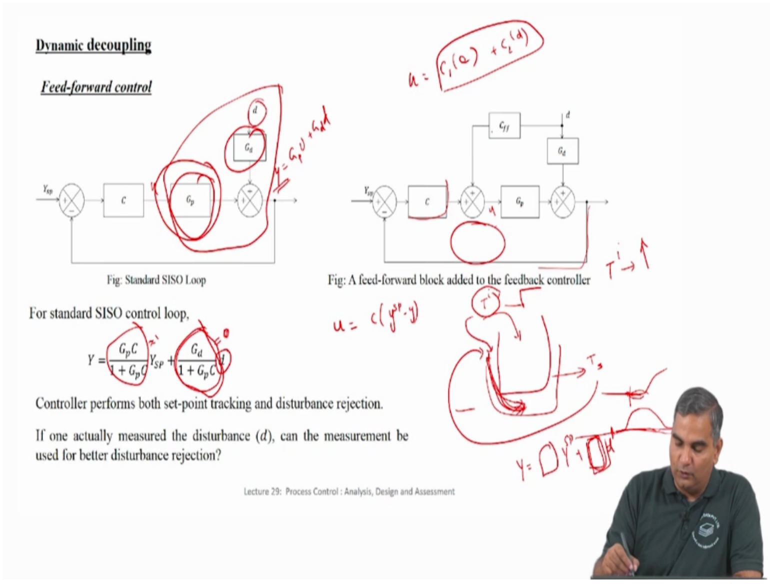
Process Control - Design, Analysis and Assessment
Professor Raghunathan Rengaswamy
Department of Chemical Engineering
Indian Institute of Technology Madras
Traditional Advance Control - Part 7

We will continue with our 29th lecture, Traditional Advanced Control, part of this course on process control. So, in the last lecture we started talking about adding more structure to the closed loop, so I started by describing a standard closed loop and then summarized all that we have learned in terms of the standard closed loop system and then I talked about adding some structure to this closed loop in terms of a cascade control.

I explain physical system, why we made need to use cascade control and then explained how you could basically modify the closed loop system by including a measurement and making hardware changes which allows one to come up with two loops within this closed loop diagram, one inner loop, one outer loop and then I also showed how you can simplify the inner loop and then come up with one block diagram which includes both the inner loop and outer loop in a form that we have seen before so that we can go ahead and do all the analysis that we have learned from this course.

Now, that was called the cascade control. In this lecture I'm going to introduce one more such idea which is called the Feed-forward control and I will show you how you look at this Feed-forward control, what kind of improvements in control that one can get by using a Feed-forward control. Over and above that this idea Feed-forward control basically introduce another concept which is cancelling the effect of a particular variable completely if it is possible and this notion is very interesting because this is a notion that we are going to use in the design of decoupling controllers which I will talk about in the lecture after this lecture.

(Refer Slide Time: 2:12)



So, let's look at our standard closed loop diagram again here and we have seen this many many times, so just one more time so that we have a context for how this closed loop diagram is going to be modified when we have Feed-forward controller included in the control design. So again standard you have Y compare it with Y set point go through a controller and input Gp and so on.

Now, clearly we said that the reasons for control again we have said this many many times but worth repeating here because we are going to do something towards disturbance rejection here. The previous cascade control case I showed you how you can kind of modify the transfer function with respect to Y set point by splitting Gp into Gp1 and Gp2 and then modifying Gp2 by a measurement to close the loop so that we have another tuning knob.

Here we are going to focus more on the disturbance transfer function and see how we can actually improve the disturbance transfer function in this Feed-forward control. So that's the slight difference between what we saw last time and and this time. Anyway, we have a Gp the controller and Gd and Y is going to be both GpU plus GdD and of course once you close the loop you can write this Y as a function of Y set point and D and in that case you will get this transfer function for set point and this transfer function for disturbance.

Now, ideally you would want this to be 1 we have said this many times and we want this to be 0, right. But clearly we cannot get zero all the time because I have a Gd transfer function Gp transfer function and the controller has its limits because we know if we push the controller beyond a certain limit we will get into instability and so on. So, remember while we write this, now when the actual controller is being implemented, notice that I don't need the value of disturbance at all for the controller to be implemented, this is just from an analysis viewpoint where we say once I implement a controller, if I want to understand the disturbance rejection characteristics, then I have to look at this transfer function form and then say how does it look, okay.

The key idea here is in a real system you are going to get a measurement, so think about what you need for controller implementation, you don't need anything other than the actual measurement, a set point value which is should be given and once you compare these to the error is sent through the controller and use computer, you don't need to know this transfer function, in fact you don't need to know this either, right.

As long as you are able to design a controller then you actually implement it on the process and the process behavior is what is going to lead me to a output which I keep looking at and doing this, okay. So, just because we write like this don't assume that we need to know D to implement a controller, that's not the…

However if there is a system where actually this D itself is measured, right, can we use that information? Right now if you take a typical controller, think about what information it is using, it is using U is some controller function which is simply using Y set point minus Y, okay. So, that is all the information that the controller is using in deciding what this U should be.

But we assume that this is how the standard control loop is, because we assume in most cases we do not measure the disturbance, we do not know what the disturbance value is. In fact we go as far as saying we do not even know how many types of disturbances are there, okay. So, it doesn't matter, the beauty of feedback controller is there might be many types of disturbances and they could take different values, we simply ignore all of this and then we simply look at driving the difference between what we want and what the actual value of the output is to 0, okay.

So, by just driving that and using a PI controller we have shown that you can actually reject the disturbance without ever measuring what the disturbance is or knowing how many types of disturbances are there and what other transfer functions and so on. So that's the beauty of feedback control in in its original sense.

However, if there is a case where you are actually measuring the disturbance, okay. Is there some way to improve the controller using that measurement? Is a question that we are asking and that's the question that we are trying to answer using Feed-forward control. Let me give you a physical example of this, if you are let's say changing controlling the temperature let say of some fluid, so you have some fluid coming in the temperature is here and let say you have a jacket where you are flowing the coolant and then you are controlling this temperature of whatever the system is the temperature of the system you are controlling.

Now, whenever the temperature changes from it set value than what you do is you go and manipulate the coolant flow rate, okay. Now, let's say there is some inlet temperature at which this fluid is coming in and outlet temperature is measured at certain value and this is the feedback loop. Now, if you never measure this inlet temperature you do not know what the variation is, basically the only information you have is this system temperature and that you use to manipulate the flow rate.

Now, imagine that everything is nice the system it at steady state so I have a certain Ti temperature, I have a certain system temperature and I have kept a certain flow rate in which the system temperature is maintained, so everything is going very well and we are at steady operation. Now, let's assume for some reason there is a disturbance (())(7:49) of this unit and because of the disturbance (())(7:52) of this unit let's say this Ti inlet temperature let say increases.

Now, let's consider to cases where I do not measure this Ti and a case where I measure this Ti, right. So, let's first take the case where do not measure this Ti. Now, if I do not measure this Ti let's say the Ti was the steady temperature, T was at this steady temperature and flow was at some steady flow. Now, the Ti has increased which I have not measured, okay. However, after this Ti has increased for Ts to show a change it's going to take some time because I have already said many of the systems have at least some little bit of time delay number one and there is also a dynamics associated with it.

So, basically if let's say this is the point at which Ti change, this is the point here, you might still not see the change in the system temperature for a while because of time delay and then maybe when it starts if it's the second order system, first order system may be when it starts changing in response to this Ti it's going to slowly change something like this, okay. So this is the point at which actually Ti change, I didn't measure it so I don't know that happen.

So, from a control viewpoint what is going to happen is I have to basically wait for this time and wait for it to increase significantly because before a significant control action is taken. So, in that sense the controller is going to be kind of sluggish because it has to wait before the effect of disturbances actually manifested in the control variable. So, this is the case where I never measure Ti.

Supposing I were to measure Ti would I want to use that information or not, one case just you say, okay, even if you measure anyways ultimately the impact of this Ti is going to be seen in Ts and once you see that impact you can actually go back and manipulate my flow rate is one philosophy and then say that's way I am going to operate. But other thing to do is to say if this Ti suddenly changes I know that this temperature is going to change down the line so can I do something in anticipation of the coming change in the system temperature, that seems to make a lot more sense.

So, if I make him have a measurement is there some way in which I can give advance warning to the controller saying look this system temperature is going to change so please start acting right now so that the impact of this change can be minimize. So, immediately since this is a change if increase the flow rate, if I have a standard feedback controller alone at this point Ti change but Ts does not know it so it starts doing this once it starts doing this the controller acts and maybe it brings it back to the set point by increasing the flow rate like this, this might be just the feedback controller.

But if at this point itself you know that the flow is going to change that the temperature is going to change, is there some way in which you can keep manipulating this flow so that this actually remains like this is a question. In other words because I know this can I use this information in such a way that this outlet temperature never changes and the disturbance is completely rejected, is it even possible is the question that we are going to ask,, okay.

So, that is the physical situation. So, the information about the disturbance measurement is there some way which I can feed it back as information to the controller so that this disturbance does not show but all that is this Ts remains at the flat value. In mathematical terms what that would mean is whatever is the though I have Y there is a transfer function for Y set point and there is a transfer function for disturbance and if the T does not change at all with respect to disturbance then basically what you are saying is this transfer function is zero.

Clearly in the standard feedback loop that is not zero it is Gd by one plus GpC. Now, by giving this information from the disturbance to the controller is it possible to make this zero is an interesting question that you can ask. Now, this physical idea of this Feed-forward controller when we put it into a block diagram again notice something interesting, this is your standard feedback loop, right, if there is no measurement of D so nothing happens here but once you have more information in the cascade control case we started measuring these temp position that is more information.

In this case we assume that the disturbance is measured, that is more information. So, whenever there is more information the key idea is how do I modify the close loop diagram to accommodate this information? So that the controller is better, the control is better is the key question that we ask in cascade control and the same question is what we asking in Feed-forward control also, without any extra information there is nothing you can do, okay.

Now, if you have extra information of course you can loop more and more sophisticated control but our first line is we have learned so much in traditional feedback control so whatever extra information is there how do I make modifications to my standard loop so that this extra information is fed into this and I am still going to analyse all of this using my standard block diagram techniques that I have learned, I don't have to learn anything extra to be able to analyse and think about these systems, okay.

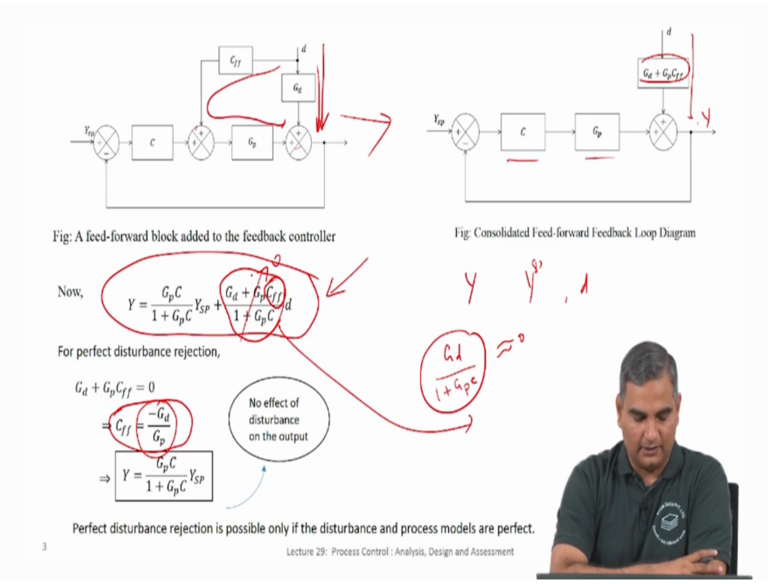
So, if we take that approach then basically here what did I say? I said the information about Ti has to go somehow to the controller and the controller output has to be somehow modified. So, the way to realize this in the block diagram by adding more texture and layer to this diagram is the following, typically if you had only the feedback error there is one block which manipulates a feedback error to give you some input this is the flow, right? So you use some control which is working on the error.

Now, I said is there some way in which I can modify the flow rate with this information that means I'm saying if I give you D also now can you add something to the U so that I kind of modify this in anticipation while in standard case I would only modify this when I see an error but I want to now modify this in anticipation of the disturbance that is coming which already have information about. So, since its linear case the standard thing to do is to look at another control aspect which will basically work with the disturbance itself and together a combination of this addition will be the actual flow rate or the actual input to the process, so that's a basic idea.

Now, if you want to do this addition then you have to have another block, so standard it's only C operating on E. Now because I have this measurement information, otherwise I cannot do this block, since I have this measurement information to U, I'm going to add this another controller so the actual input the process in terms of the manipulated variable has two terms, one term is based on the error that I see, the other term is based on the disturbance that I know is happening, okay. So, this is the way in which I modify this block diagram.

Now, if I modify this block diagram this way then basically I have to do a simple modification of this so that it comes back into the same form as here. In the previous case there was a change within the loop, feedback loop in this case actually the change is coming from outside the loop.

(Refer Slide Time: 15:36)



So, if you do a simple manipulation of this and then say how do I convert this to my standard block diagram, the standard block diagram will be simply this. Now, notice that there is no loop that this participates in, so basically all is happening here is there is a d times so d would have generally gone through only Gd and then affected Y but now d is also going through Cff and that Cff is getting multiplied by Gp so this is going to have one term coming from this direction which is Gd time d and another term which is going to come through this which is Cff times Gp.

So, if you look at this, this d initially was affecting only through this path, now this is going to add and then affect through this path and also this path. So, this structure when you modified it and put it in the standard form it will look something like this, the C will be there, you're Gp will be there, however the disturbance is going to affect through Gd and one plus GpCff, okay.

Now, the fact that this is part of this loop will come out when you do the overall transfer function which we will see. Now, you have this, right. Now, it is in standard form, now if you want to write the transfer function between Y **Y** set point and disturbance you use the standard form, now instead of Gd you're going to use Gd plus GpCff. So, this is a simple computation and you can actually write all the equation corresponding to this and also derive this, okay.

If you are not comfortable with simply writing this as Gd plus GpCff buy this argument, what you can do is you can write an equation for this node, you can write an equation for this node, you can write an equation for this node and simplify all of this, you will get this, okay.

However if you understand this and put this here then you do not have to write this equations, you simply say the transfer function between Y and Y **Y** set point is go all through this and then I get GpC divided by 1 plus GpC and the transfer function between this and this is Gd which is now Gd plus GpCff divided by 1 plus GpC the same role that you have, so this you can easily verify for yourself that this is actually what happens.

Now, something interesting has happened, remember in the cascade I said somehow if you had the just the standard feedback loop you have to change your hardware in such a way that some part of the transfer function gets change. So, in that case we had Gp to Gp1 as Gp and then we showed that the Gp2 could be converted to Gp2C2 divided by 1 plus Gp2C2, right.

In this case because we are talking about measurement for disturbance and using that measurement in a controller we are looking at changing the Gd itself and by adding this Feed-forward loop that is Feed-forward meaning based on the disturbance value itself I'm asking the controller to do something and not wait for the error, that's the Feed-forward loop. I get a Gd which is now different which is Gd by GpCff in the open loop and the closed loop transfer function is Gd plus GpCff divided by 1 plus GpC, okay.

Now, I have a tuning knob which is this, okay. So the question that you can ask now is remember in the previous slide I said can I use this information about disturbance to ensure that output is not at all affected by d if I know what the value of d is, so if I don't want the output to be affected by d at all our goal has always been to say that our most ideal transfer function is Gd by 1 plus GpC zero but we cannot get this, right because we had no tuning other than the C and you can get this to 0 by only making C bigger and bigger and bigger so that the denominator is a very large value and this goes to 0, but we have already discussed this we said that because of the stability limitations and so on if you make C very large you will become unstable, so you cannot do that, you cannot get your ideal value.

However, now with this equation now we have that chance because I have somehow introduced a tuning knob in the numerator, so do I do not have to worry about C, there is another way to get this to 0 which to make the numerator zero so if I say Gd plus GpCff is zero, Cff is minus Gd by Gp. So, if I tune a controller such that this is minus Gd by Gp , then it looks like I can make this term go to 0, that means this is the first time I am seeing a controller which is perfectly disturbance rejecting.

So, this is absolutely zero, there is no dynamics, if you give me the disturbance value I can immediately come up with a controller which never waver from a set point because of disturbances. It will waver one from a set point only when you actually change the set point and asked the controller to do servo control.

So that's a fascinating idea and the for the very first time I'm showing you case where you can actually make a transfer function go to 0 by setting a numerator to 0. However, just remember a few things in reality if you design a controller like this the disturbance would not be perfectly rejected because number one all of this is ultimately when we are doing this controller design and so on this are all model, so right, this is a model for disturbances, this is a model for process and so on.

So, once you design the controller when you actually put it into a closed loop this will be the true process model, true process transfer function and this will be the true disturbance transfer function which can be different from the model. So, since this are all defined and derived from the model just like how we talked about direct synthesis, in actual situation we have two use transfer function with our which are different from the model to see the effect or when you actually implement it they will turn out to be different so that there won't be perfect rejection, so that's number one that you have to remember.

The other thing you have to remember is that we typically want transfer functions where the denominator polynomial is of a higher order than the numerator polynomial in some cases that might not happen when you have a transfer function like this so you might have to add a certain filters and before Cff which will make not go exactly to 0 but it will still give you very very good results, right.

So, from a practical implementation viewpoint since we're using models while we can design the controller the actual performance in terms of disturbance rejection might not be perfect but it will be still much better than not using this Feed-forward loop when disturbances are measured. Of course if disturbance are not measured we do not have this option of measure of doing this Feed-forward controller itself. And also remember that from realize ability point of view and from other considerations we might not simply use Gd by Gp we might actually use a filter in front of it so on, so that the kind of moderate for other reasons in terms of the controller design, okay.

But nonetheless this is the first time I'm showing you an idea of actually rejecting the disturbance completely by setting a transfer function to zero through measurement of disturbance, okay. So, in the sense it's a new idea for this course. Now, the interesting thing is we are going to use exactly the same idea to now look at let's say two control loops together and how we use this idea to make a MIMO controller with two loops or two variables, two inputs and two outputs to behave like SISO controller by simply using this notion of rejecting this disturbance, okay.

So, the idea is going to be which I will describe in the next lecture if you think of two loops, U1, U2 Y1, Y2 whenever you change U1 Y2 also changes so we can think of U1 as the disturbance for Y1 and similarly you can think of U2 as a disturbance for Y2, is there some way in which we can do good control while rejecting the disturbance effects that comes from the other loop.

So, if you think of what is happening in the other loop as introducing disturbances in this loop is there some way in which I can nullify that effect using very similar ideas in terms of setting some transfer functions to 0. So, that is going to be a logical extension of this Feed-forward controller which is called a decoupling controller and that's the first time we are going to actually look at linear multivariable systems and using some of these ideas to reduce that linear multivariable system to decouple single input, single output systems.

In that we are logically going from basic closed loop to adding more texture and structure to this closed loop using more blocks based on information. In cascade control we have the stem position information, in Feed-forward control we have the disturbance information. Now that we have learned some of this the next logical step is to really look at actually a multivariable system and I'm going to do a 2 by 2 and that idea is very easily extendable to 3 by 3, 4 by 4 and so on. And then we ask the question is there something that we can do, control this 2 by 2 system using SISO or single loop ideas, so what other things we might need to do that and I'm going to show you how this idea that we saw in Feed-forward can be translated to decoupling controllers, dynamic decoupling controllers, we will do one lecture on that to give you the next level of thinking in terms of multivariable controller. So, I will see you in the next lecture, thank you.