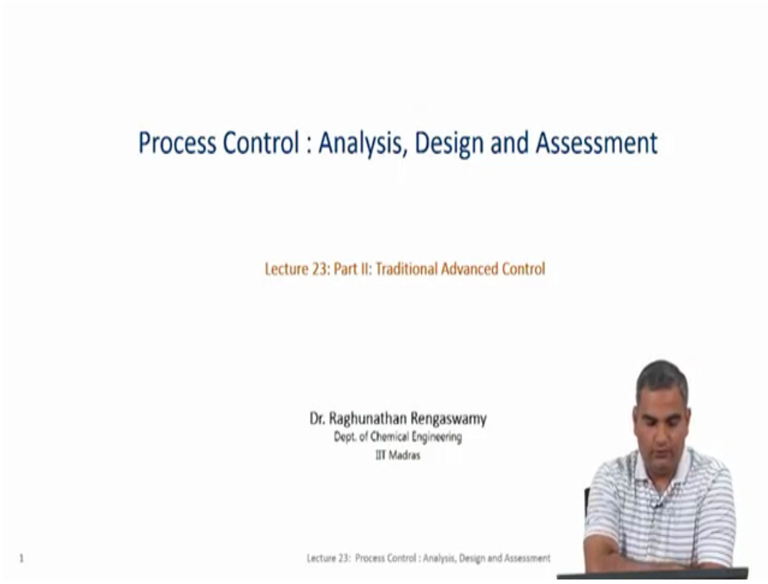
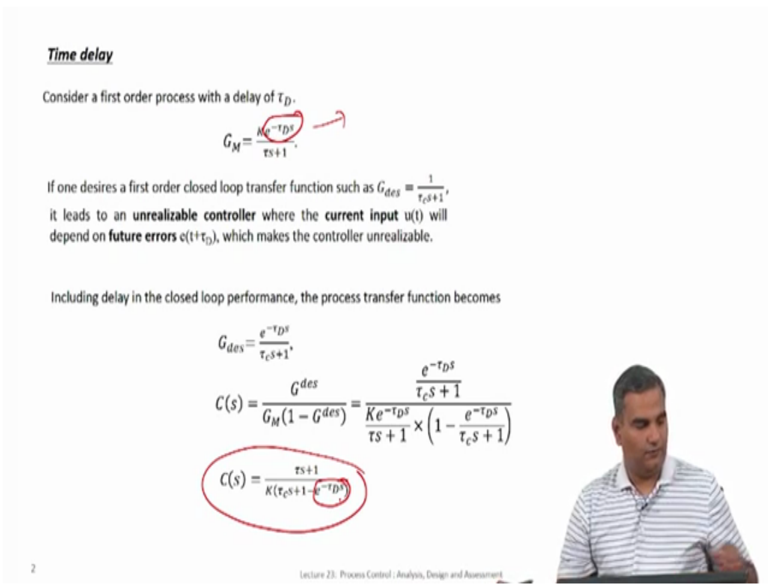
Process Control - Design, Analysis and Assessment
Professor Raghunathan Rengaswamy
Department of Chemical Engineering
Indian Institute of Technology, Madras
Traditional Advanced Control – Part 5

(Refer Slide Time: 0:14)



Let us continue with our lectures on design of controllers for time delay systems.

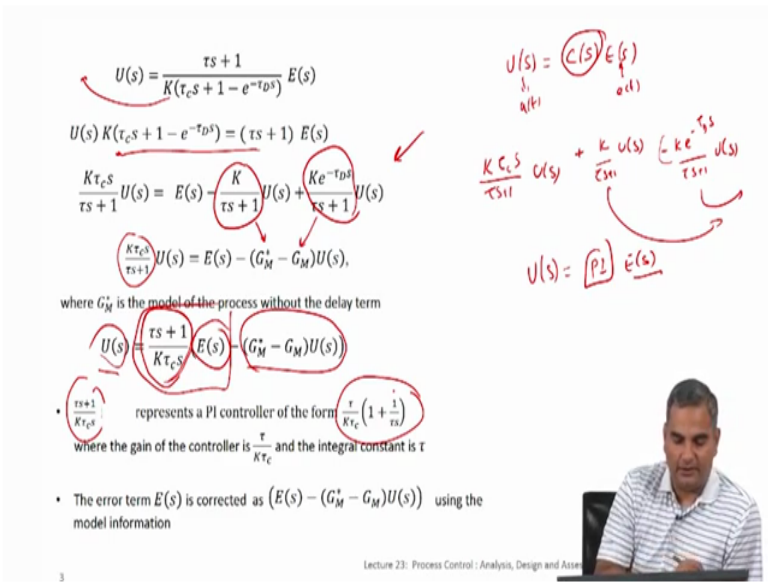
(Refer Slide Time: 0:20)



In the last lecture we talked about how we could start with this controller design which we have seen several times before and then convert this to a polynomial form numerator or denominator using Pade approximation, then the controller becomes your numerator by denominator. Another way of saying the same thing is to basically say here itself convert this into a numerator by denominator form through Pade approximation and then you will get the process itself as a numerator by denominator or the model as a numerator by denominator, then you use these standard techniques that we have taught to understand controller design okay. So this is where we had stopped in the last lecture.

What I am going to do in this lecture is I am going to look at other ways of thinking about this problem where we do not use approximation such as numerator by denominator polynomial either for the e power minus tau D s or in general, whether we can handle this term as it is and then understand what are the implications of this term the denominator and how do we go ahead and design our controller for time delay systems.

(Refer Slide Time: 1:32)



So what we are going to do is we are going to start with some very simple algebraic manipulations and then I am going to derive some simple equations and then we will look at ultimately how the controller can be implemented without ever using the Pade approx., so is that is a goal of this slide. So as I said before as we go into more and more advanced control concepts the key difference that you will see is that I have talked about this before also just to reiterate the the notion of the notion of the model itself in the controller design and the controller computations will become more and more, in a PID controller the notion of the model the controller design exists but it is right at the beginning where we fix the parameters of the controller using a notion of a model.

Once that is done then in the computations and so on the model never comes back again it is forgotten and gone, controller design now it is controller tuning parameters, the controller works. However, in direct synthesis controller idea the model is more involved in the sense that the controller design equations now like CS itself directly uses the transfer function model in its full detail right.

So we have this 1 over GM G (())(2:56) by 1 minus (())(2:58) so the GM which is a full model for the process is directly used in the controller design equation itself. So understand the difference between this and the PID controller, where of course the model information is used but the model information is used indirectly to fix the controller parameters, however in the direct synthesis approach the model transfer function actually takes part in the definition of the controller C of S.

However, after that is done while you are computing the control move so the transition from C of S to control move is actually assumed that you have designed a controller you have a controller design and then the equation that is important is C of S times C of S which is equal to U of S. Now here is where once the controller is designed you actually use it in online as soon as an error term comes in then the controller takes an action. So this is basically the u of t comes out of this and u of t gets activated once I have an error.

So even in direct synthesis approach while the model form explicitly comes in the controller definition structure definition it does not come back again while implementing. So while implementing I do not worry about the model because I already have a controller structure I need only the error and then I can keep doing this. However, even more advanced controllers and the first of those that you are going to see is the controller for time delay more than even being a part of the controller definition in terms of the model transfer function itself being used even in online computations while the controller is acting you will need the model in this system. So that is the next level of the model being a part of controller implementation.

So if you think of controller design and implementation the simplest is controller design where the parameters are dictated by what model you have, the next level is the control structure itself is exactly defined by the model transfer function and the third level is the controller design is dictated by the model transfer function and on top of it while implementing you would still need a model.

So we are going to see that idea for the first time when we try to solve this time delay controller problem without using an approximation like Pade approximation, if I use the approximation like Pade approximation it is at the level of the controller model coming in as its whole form in the controller definition that is where it stops but if I do not do Pade approximation you will see something very interesting here okay, so that is the thing that I want you to understand.

Now if I have this equation which is U of S C of S by E of S and I am going to say I am going to deal with this e power minus tau D S as it is. So let me try and start doing something interesting. So what I do is I take this term to this side and multiply U of S by that, so I have U of S times K tau C S plus 1 minus e power tau D S is tau S plus 1, then what I do is I push this back to here.

So then I will get the first term will be I will have K tau C S divided by tau S plus 1 U of S that will be the first term, the second term will be K by tau S plus 1 U of S and the last term on the left hand side will be minus K e power minus tau D S by tau S plus 1 U of S so these are the terms that will be in the left hand side and on the right hand side I will have only e of s.

Now if you move this and this term to the right hand side you will get this equation the first term K tau C S plus tau by tau S plus 1 U of S stays here, I have E of S the plus became minus here, minus K tau S plus 1 U of S and the minus became plus here, this minus here became plus here when I move to the right hand side, I have plus K e power minus tau D S over tau S plus 1 U of S.

Now as soon as you look at this equation you will see something interesting which is I have this U of S term here typically like I said before U of S term will have some PI related terms any E of S but it looks like on the right hand side there are other terms that have come in which are not only something that is due to the error. So basically I can combine these two terms and then I can say keep the left hand side as it is K tau C S by tau S plus 1 U of S is equal to E of S minus taken a minus out I am calling this as GM star and this is GM times U of S.

So basically what I am saying here is that if you look at this this is actually the model itself G model itself for this process and this part there is only one thing that is missing which is this e power minus tau D S. So let us call that G M star that is this is a model of the process if there had been no time delay okay so that is the interpretation for GM star. So the model of the process with the time delay is K E power minus tau D S by tau S plus 1 and if you talk about a model of the process without that time delay then I will have K or tau s plus 1, so it looks like this.

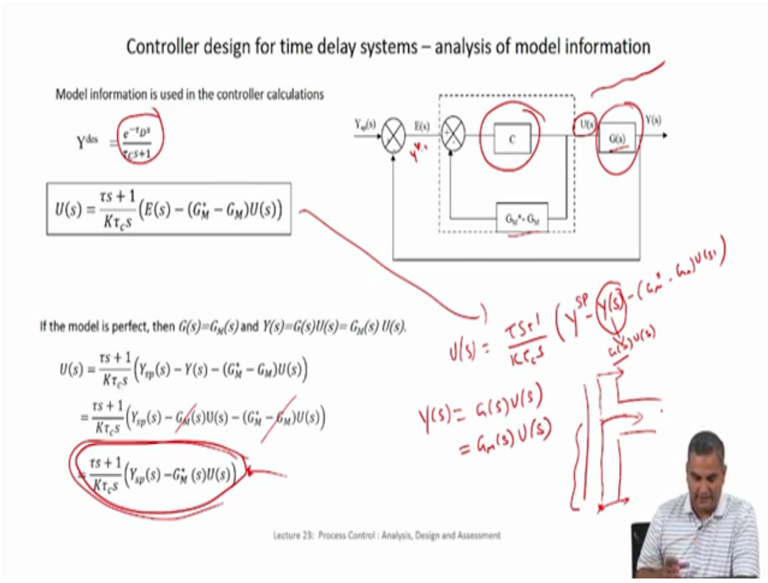
Now what I can do is I can take it take this term to the right hand side and then combine all of this. Now you will notice we have said this several times if you look at this tau S plus 1 by K tau C S this is basically the PI controller okay. So the PI controller is typically going to act on only error if there are no complicated dynamics, but for the first time what we are saying is this PI controller is going to act on not only the error but also some prediction term that has to use the model so that is the first time you are saying.

So even in implementation it is not enough if I just get the error term, I have to do some processing of the error and the processing of the error I am going to do by subtracting another term and the other term is actually going to use the model itself okay. So if I had stopped here then the effect of the model is done PI I get the error I implement the controller, but for the first time when I implement the controller and I have to compute a u of t I cannot just live with my error, I am also going to do some prediction with the model, so model predictions become part of implementation okay.

So this is the first time you are seeing this and this concept when you take it to the next level is when you will you will see a model predictive controller so model is predicting something and those predictions have to be used in the controller computation and the simplest form of that you are seeing here because we are seeing the controller computation requires the model and as I said before this is very easy to see why it is the PI form we have done this several times. So if you take this and then you do this you will get the PI form, so this and this are the same and the PI form is K C times 1 plus 1 over tau S, so you get the PI form.

So now suddenly you notice that you can actually implement the controller without any approximation for E power minus tau D S because we have not approximated it. However, the price you pay for that is you have to actively use the model in a prediction mode even during controller implementation, so that is what this equation says okay.

(Refer Slide Time: 10:54)



So in a real implementation, so I just want to interpret this in terms of what will happen actually when a controller is being implemented using this, so what basically it says is the following. So you get an error so let us assume this is actual process and this is the actual process transfer function so you can think of this as a tank or a reactor whatever it is which is the process itself and then all of this is the corresponding electronics and the control system associated with it and this is how this block diagram will be implemented basically whenever I get an output value which is from the true process it is compared with the set point which generates an error.

Now based on the U that you had or you have implemented what you are going to do is take those and then actually predict some values using this GM star minus GM block function and then the error that you get which is Y set point minus Y you are going to correct with this prediction and that is what will go into the controller and that will give you U of S which will be implemented for you to get Y of S, okay so this is what will happen actually okay.

Now it looks like you need U of S to compute U of S that will be derived very simply in the controller computations for now let us just think about if I were to make this U of S move let us say if I (make that) take that into consideration then I will have to see what will happen through this prediction and that prediction will be somehow used to correct whatever error term I have and that is the new error term that will go into the controller which will decide what the U is and that U will be implemented and then you will go through this. So there is this notion of prediction that we are seeing for the first time in controller implementation.

Now if as I said before again the other thing you have to notice is look at this in this transfer function block for the first time you are seeing G which is actually representing the two process and the model which is actually our conceptualization of what this is and while this model has been used in the controller computation itself, it is also being actively used in the loop while the the controller is being implemented, so that is the new idea that we have here.

Now if it turns out that the model and the process are perfect let us say G model G process and they are exactly the same I want to know what happens to this equation, right now we have not made that assumption because you also have to remember this U of S is tau S plus 1 by K tau C S, E of S is actually Y set point minus Y of S okay, then we have this term minus GM star minus GM times U of S okay.

So the Y of S is actually coming out of the process okay so it is not coming out of the model the assumption is that it is coming out of the process, so this Y of S is really G of S U of S okay. So while in the controller we do not have to worry about this G of S because we directly get Y of S as far as the computation is concerned all I need for the controller to be implemented is actual measurement from the plant, the set point and the model and the model could be anything right and it could be exactly the process, slightly different and so on and also keep in mind that this is that (())(14:21) sorry.

Now I might ask this question if the model and the process are exactly the same what is this representing. So can I understand this equation and get a little more insight into this equation by understanding what happens if the model and process are exactly the same in which case what we are going to do is this G of S is exactly GM of S right. So if I do this Y set point I have Y of S GM of S times U of S which is exactly what it should be then what will happen is that this is minus GM S U of S, this is plus GM S U of S Plus and Plus that and that will get cancelled so what I will be left will be just Y set point of S minus GM star U of S so this is what I will be left with if Y of S equal to G of S U of S is exactly equal to GM of S U of S, so this is an important thing to notice.

So once I have that then these two terms will get cancelled and what I will be left with this Y set point minus GM star U of S. Now if the model and process are exactly the same then basically what this controller says, another interpretation this is a little more difficult to kind of conceptualize the first step as you do more and more work in this area this will become a little more obvious, but if you look at this what it says is the controller move could be such that you compare this Y and a model where the model has no time delay.

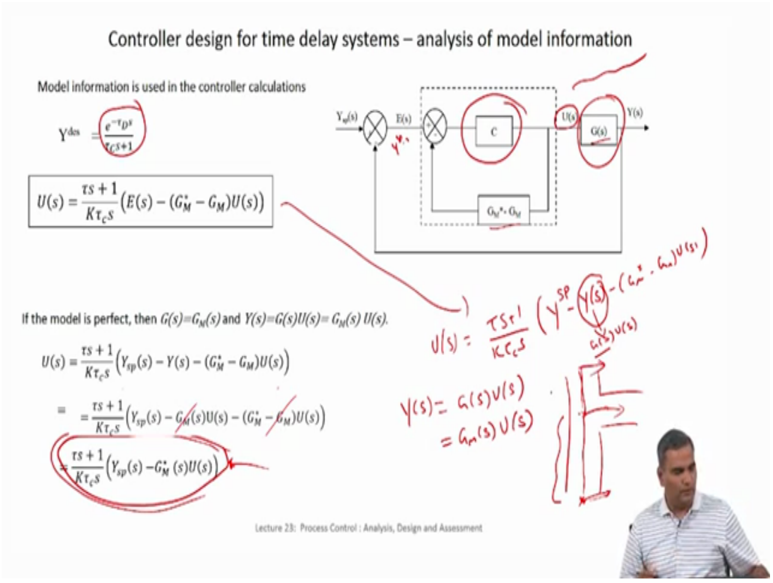
So basically somehow you factor the model time delay outside of the controller and then make your decisions as if that there is no time delay and then live with your decision and the time delay will automatically show up outside of this loop okay, so that is what this seems to say. So this would be a great controller if for example the true process and the model of the same and on top of it the model did not have any time delay also. So this is basically saying define a controller as if you are designing this controller for a process without time delay, take your control action and wait and the time delay will automatically come into picture outside of this loop.

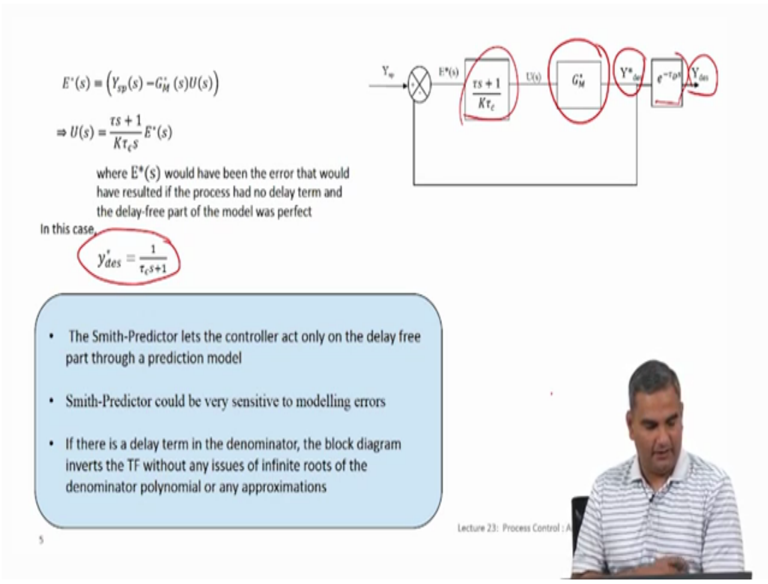
So if you want to understand this based on the bike example that we talked about before, now if you are going on a road and then let us say you have to take a right turn after a little bit of time and let us say you exactly know what your time delay of your process is let us say it is 5 seconds, then what are you going to do? You know that at some point before the road, so if I have a road I have to take a right turn I am here and I have to go here however if I go here and then take a right then basically because of the time delay I am going to go a little far and I am going to go and hit the road here right so that is not what I want.

So basically when I am sitting here logically the best thing to do is if I am able to estimate that the time delay is going to be this much, then when I want to turn right here basically I am going to turn right here okay. So what this is is exactly what this equation is saying. So I am going to make a decision that I would have made if there was no time delay that is the reason why I took this right here itself okay, so that what this equation says.

So once I make that decision then the time delay will automatically make everything perfect because I hold my steering position like this I keep going at this point I will turn right okay. So this beautifully explains that the best thing to do is to take a control action as if there is no time delay to your process, if I turned right I will go right okay so like that I take this decision and then I simply wait and then I will go and get to my correct action that I want. So you can see how the mathematics and the intuition what you do all of them come together nicely when we think about this time delay systems.

(Refer Slide Time: 18:40)





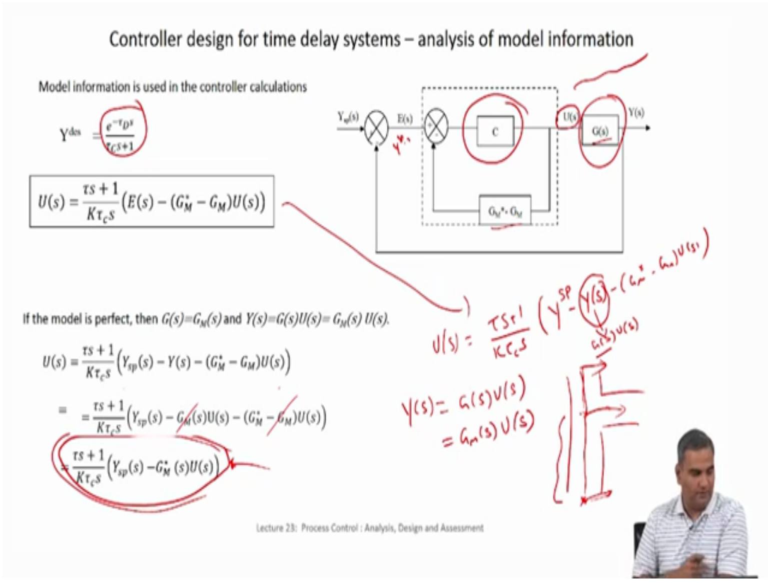
So which is basically what this block diagram is saying, it is saying okay keep your (()) (18:45) as if there is no time delay, have a model as if there is no time delay so in the previous bike example Y decide is I want to immediately turn right, that is what I am saying though I know it will take a time but my controller computations are assuming I want to turn, in immediately right there is no time delay here, so you give a PI controller it will make me do my right turn okay.

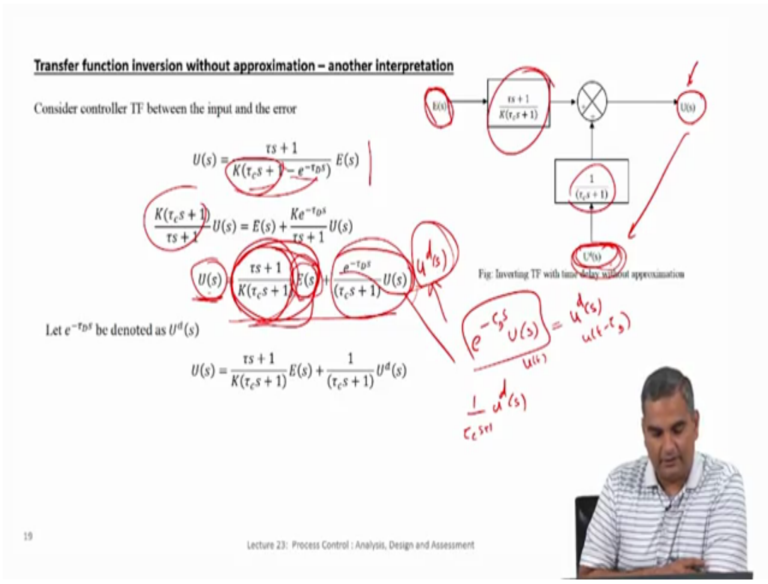
So this is notional but the process itself has a delay so it will take that amount of time to actually make the right right so then you if everything will be all right you would have made the correct right turn you will not go and hit the pavement and so on so that is the idea of this controller. So basically this before all of this explanation was given and before all of this ideas of direct synthesis and so on were out there this kind of controller for time delay systems was called the Smith predictor which basically predicts the response without the time delay and basically says okay just do the controller design based on that and everything will be okay so that was intuitively direct which can be explained very very nicely using the direct synthesis idea in terms of how all of this comes together.

However, if there is the model time delay if very different from the process time delay and so on then there are lots of issues with the Smith predictor controller or even the time delay equations that we derive you can see that right because if I were to go back to this example and then say well it is going to take me 5 seconds to get here so 5 seconds prior to that I have made change here and I computed this I should make this change here 5 seconds before because I am assumed it will time delays 5 seconds so based on that I figured out how long it will take me to go from here to here, so when I am at a position where I think it will take me 5 seconds to get here I make this move.

But if the time delay were actually 8 seconds then you are seriously in trouble because you made a decision here saying I will come here in 5 seconds so I am going to turned right but you have come here but you are still not turning right because the actual time delay is 8 seconds so you will go further and do the same thing, so you can see why this becomes very difficult problem, so there are ways to deal with this but it is important to understand the difficulty with this. However, if you have a reasonable idea of a time delay then this is a wonderful way of controlling time delay systems.

(Refer Slide Time: 21:20)





Now we know how to think about a controller for time delay systems we also know how to interpret the controller we got which is basically to say assume that there is no time delay and take an action as if there is no time delay based on whatever G decide you want and simply wait and things will be okay right. So we have done all of this, so the one last part that we need to understand is how how do I look at the stability of the system because all of this is great I have this equation and so on but I cannot use now standard notions of stability at all right because the standard notion of stability I needed a numerator by denominator polynomial, I either figure out the roots of the denominator polynomial and then look at if all of these are in the left half plane or if I cannot write the denominator in a root resolve form because there are controller parameters which are part of the denominator polynomial then basically I use Routh table.

But look at this control here right so this is it is a very complicated controller because if you do GPGC by 1 plus GPGC then there this model part and it is not very clear how you address all of this in figuring out what your stability will be and this is actually quite nicely understood if we provide another interpretation a slightly different interpretation of the same equation okay.

So let us take the same equation and we will do instead of expanding it to think the last time we did about three terms on the left hand side so that we had over all four terms, if we expand it to just two terms on the left hand side, so that overall you have three terms you will get another interpretation of this time delay transfer function and you will see why it is going to be difficult to understand the stability of the time delay systems, clearly from purely transfer function form I have already explained because the e power minus tau D S is not in the polynomial form, all the stability ideas that we have learnt always require this denominator polynomial so those are all outside the window right.

However, we did some nice mathematical jugglery to implement the controller, now we know how to implement the controller in two ways, one is an approximate implementation if I were to convert this e power minus tau D S into Pade approximation and we also showed that that is not needed if you are willing to move on to the next idea and control which is to use the model explicitly while the control computations are being made, then we could do the same thing using Smith predictor and I explained all of those.

So implementation we have somehow managed or we have brought a new idea which is use of model explicitly in the controller computation to handle time delay systems, but that does not let us off the hook in terms of still figuring out the stability of this system, so the stability is a different question. So if you look at this and ask why is this stability a problem? So typically what we look at is if I have an input, output system I ask this question if the input is bounded will the output be bounded and so on.

So if you look at this equation you are going to see something interesting here so if I take this and then again move this to the right hand side and bring tau S plus 1 back so I am going to keep these two terms and together and then I am going to have one other term here, so on the left hand side I am going to get two terms which is one for this part and one for this part and this part I am going to move to the right hand side, so if you do the simple algebra you will get this K tau C S plus 1 by tau S plus 1 times U of S is Z of S Plus this.

Now if I move all of this to the right hand side I will get an equation of this form okay. So if you notice this the current U of S is a function of the error and you notice this right here, so this term and we are going to use the trick that we have been looking at. So e power minus tau D S U of S okay if I call this as some you delay of S so basically you recognize that whenever I have term like this in the time domain u t I have to replace by u t minus tau D okay, so that is the idea so once you get this very clearly imprinted in your thinking process then you will see that whenever that e power minus tau D S comes in the numerator it is really not a big difficulty okay.

So basically this actually tells you how nicely you can implement it because that block diagram with G model and GM star it look like you need U of S to implement U of S but if you look at this equation it more clearly tells you what is happening. So basically I can say these terms together I am going to say u d of S. So basically when I want to implement U of S okay I need the error and then I process it through a transfer function to get this term figured out and when I come here basically what it says is I have this so I am writing this as 1 over tau C of S plus 1 u d of S okay.

So here I am going to process a time delayed version of U of S with the transfer function. So if I want to compute the current you I am going to use the current error and process it through transfer function and I am going to take a time delayed version of my input and process that through a transfer function and that will also add on to what the current U should be. So now you see that it is easily implementable because if the time delay is (three) five sampling instances let us say then to compute the current U value you have to use this current error and also you have to use five time delay go back and then use the other U before that.

So when you want to compute this U basically this information already exists because this is previous U which is already been implemented, so there is no complication in being in implementing this equation and then getting a value for the current use. So this clearly tells you that the implementation is basically the current error and you have to process previous inputs that you have taken.

However, if you look at this if this block diagram then you say okay there is a U of S which is going to be a function of E of S and this u d of S, so if I want to see whether this will ever go unstable. For example is it going to go to really large values okay let us assume for a second just for sake of argument let us say this is not going to go very large okay then you can say okay this is a stable transfer function, so this additive part of U of S is never going to go crazy right this is just very simple way of thinking about this and seeing why time delay brings in all kinds of difficulties.

However this is also adding on to it so basically you have to say this if I want to make a similar argument if you say well if this does not go unbounded then this is stable so this additive part also will be stable but there is a problem because this is the same as this variable except time delay that is it. So to find out the stability of this I have to make some assumptions about a time delay version of that which makes no sense at all, so for finding the stability of this I have to assume the stability of the same variable kind of circular logic comes here.

So that is the reason why the notion of stability when I have time delay systems is a little more complicated and you need more sophisticated techniques for identifying stability of time delay systems okay. So I hope that you got a good feel for the interesting ideas in terms of how we are going to use the model itself in the controller implementation or think of it another way how we are going to use the time delayed version of the input itself in the controller implementation, which is quite possible logically.

However when it comes to stability analysis if I want to find the stability of U I have to make some assumptions about the stability of u d which does not make sense it looks like a circular argument. So how do we address this problem and still be able to talk about the stability of time delay systems is the last thing we need to understand when we talk about control of time delay systems and I will discuss this in the next lecture, thank you.