NPTEL

NPTEL ONLINE COURSE

CH5230: SYSTEM IDENTIFICATION

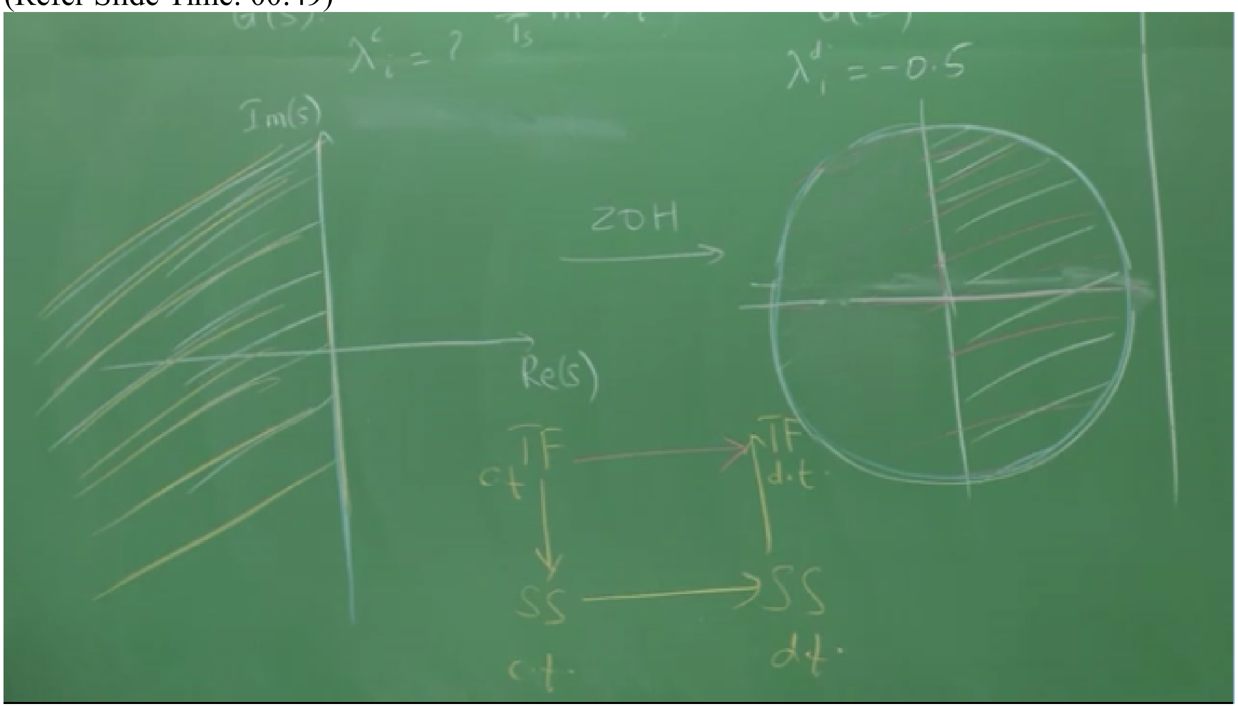
SAMPLED: DATA SYSTEM 4

ARUN K. TANGIRALA

DEPARTMENT OF CHEMICAL ENGINEERING,
IIT MADRAS

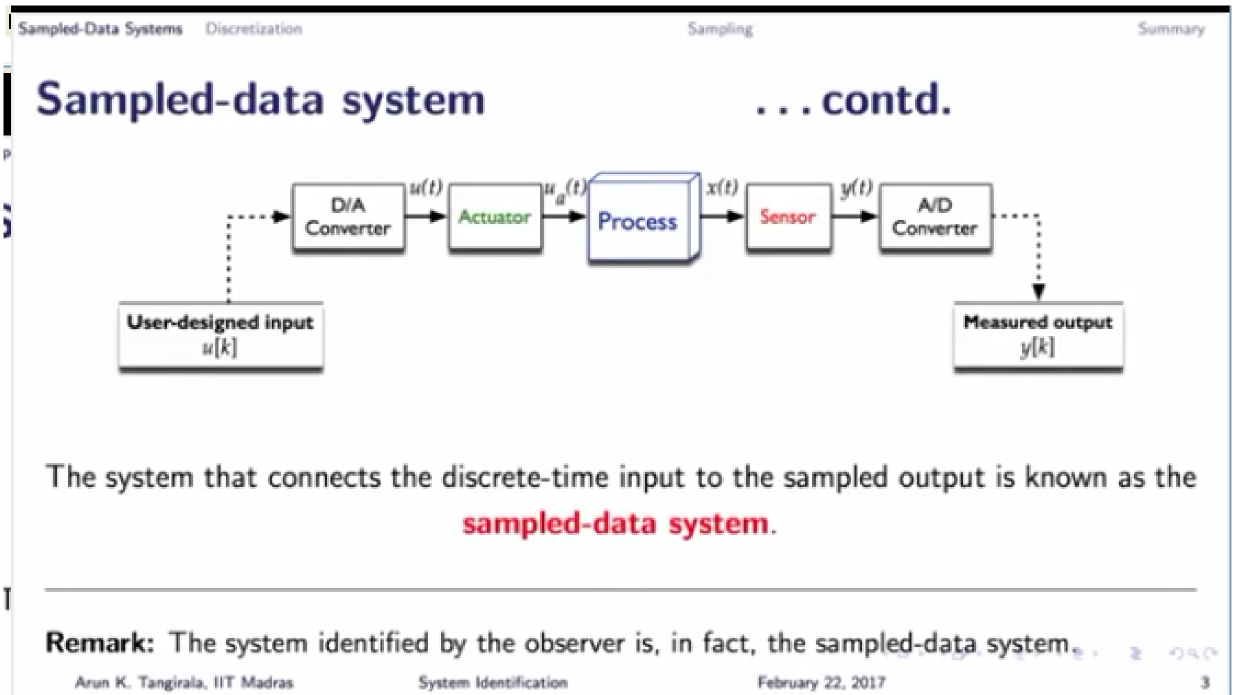
Now let us move on to learning exactly how to go from transfer function to transfer function, this is one way in fact, already now you've learned, I'll give you any transfer function at least with real polls you can break them up into first orders, applied a simple rule and then be done with it, however it is still advantageous to know how to directly arrive at the transfer function, given the transfer function description without going through this circuit as root, we didn't go through a circuit as root there but that possibility exists.

(Refer Slide Time: 00:49)



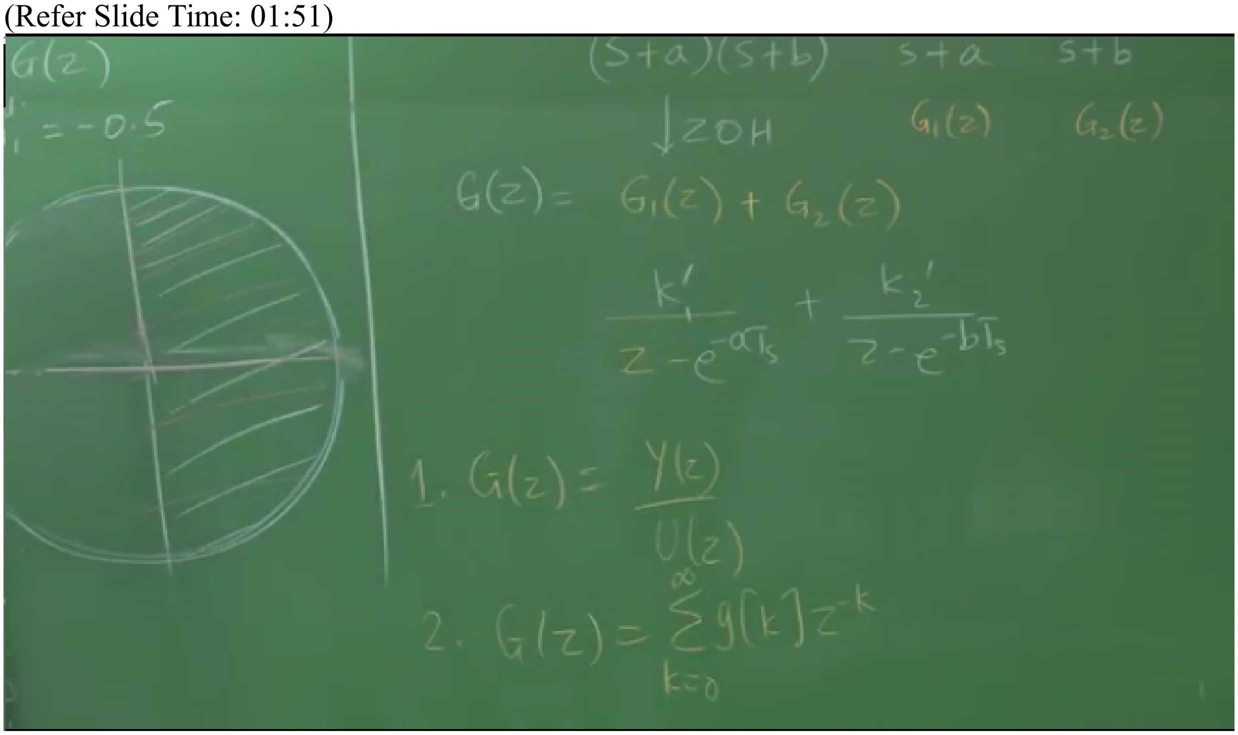
So how do we do this? What is a basic idea behind this approach? It's a very simple approach. Let me take you back to the schematic, first of all let's look at the schematic and then recall the definition, so this is the discretize system that we are looking at,

(Refer Slide Time: 01:12)



what is a definition of G(z) theoretically we have discussed right, what is a definition? One definition of G(z) is that is, so Y(z)/U(z), what is the other definition? Correct, so Z transform of the impulse response, both are valid, we'll restrict ourselves to causal systems, okay.

(Refer Slide Time: 01:51)



I can use any of these definition now to arrive at G(z) given G of us, how do I do that? Let's look at a second approach and then we'll talk of the more generic first approach, so the second approach says if the impulse response of the discrete time system is given, all I have to do is take the Z transform of it and that I get G(z), correct.

Now let's look at the system here, the system here will tell us what will be the impulse response, so what is impulse response by the way definition, what is it definition of it? What is a definition of impulse response? Response to an impulse input, correct, so I start from the left hand side here, and I have to give an impulse input, and by impulse here we mean chronicle delta not the direct delta, because we are looking at discrete time, so I give a unit impulse here, now I walk through, I walk through the system and figure out what the response is, correct, we ignore anywhere actuator in the sensor dynamics for now, what does the whole device do to this impulse? What kind of U(t) does it produce? What is it called? Step, you don't say step for a short period, pulse, so ZOH removes the I and M from the impulse and produce the pulse, impulse says I'm pulse, ZOH says I know you're pulse therefore I'm going to take your IM out of it, okay.

So what you're going to get is a pulse, what will be the duration of the pulse? Sampling interval, good, so which means now I know when I feed a pulse in the discrete, sorry impulse in the discrete time it translates to a pulse in U(t), in continuous time, and since we ignore actuator dynamics we assume that this pulse straight away goes into the process.

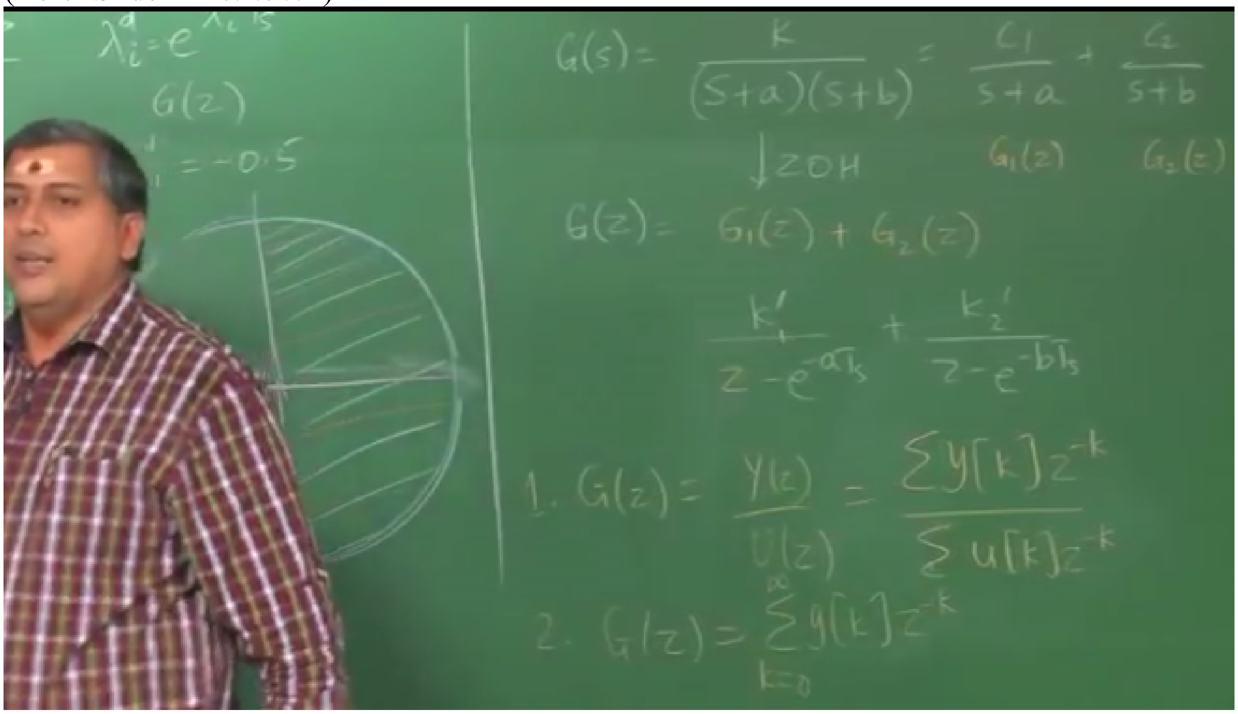
Now process transfer function is given G(s) is given, right, all I have to now determine is the pulse response, right, and that will be my Y(t) that's fairly easy, how do I determine pulse response? A pulse can be thoughts of as a difference of two steps separated apart by sampling interval, so which means I just have to take the step response of the continuous time system, delay it by the sampling interval and subtract them too, that will give me the pulse response, right, and that I can do given G(s), once I obtain that all I have to do is to obtain the impulse response I have to just sample it, that's what the sampler is going to do, then I have Y(k) which is the impulse response of the discrete time system, there is a question which is asking you to prove that the discretized system that is a Z wise discretized system is not impulse invariant. What is that mean?

Now recall the discussion that we just had, we fed an impulse, but what did the continuous time process see? A pulse, and it responded to a pulse, it didn't response an impulse, and what you have sampled is, sampled the pulse response of the continuous time system, that's okay, to the observer what is happening inside the box it doesn't matter, I fed an impulse I got a response, that is your impulse response, but is that the same as sampling the impulse response of the continuous time system. What do you think? Just recall the discussion that we just had, question is whether the impulse response of the discretized system is it the same as sample version of the impulse response of continuous time system, yes or no? Because that were to be case then the impulse should have translated to an impulse, what is happening is the impulse response of the discretized system is in fact sample version of the pulse response, and that is why we say it is not impulse invariant, right.

When we say invariance, what that invariance means is whether I observe directly in the discrete time or go to continuous time system and sample it I should get the same values at the sample instance, that is what we mean by invariance here, and that won't happen for sure, therefore any ZOH discretization is not impulse invariant, but that doesn't matter as far as transfer function derivation is concern, so long as you have correctly derived G(k), given the transfer function G(s) I can arrive at analytical expression for G(k), and then compute G(z) you may think this is a laborious procedure, the answer is yes it's a bit laborious to derive G(z) through this root, a better approach is based on this definition, okay.

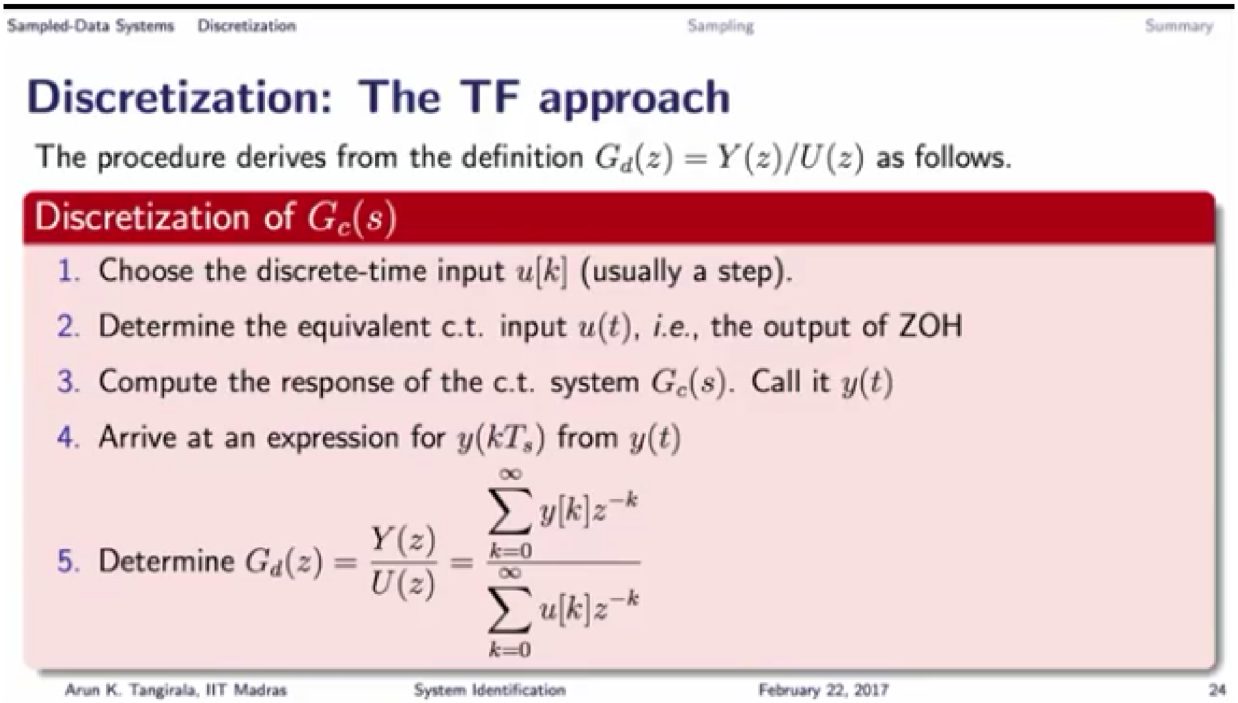
Going back to that definition and exploiting the step invariance property of GOH discretization we can arrive at a simpler method for arriving at G(z), both eventually give you the same answer, but there is ample scope for making errors and getting confused in the second approach that we just discussed, so how do we do this? Same story the idea is the same, the only difference is I'm going to replace the impulse with a step, right, so that when I feed a discrete time step to the whole it will produce a continuous time step and then all I have to do is determine step response or the process given G(s) sample a step response I get the step response of the discrete time system, which means now I can calculate the numerator here Y(k), I mean the Z transform in the numerator and since we already know the U to be a step, I can replace the denominator here with 1/1-Z inverse,

(Refer Slide Time: 09:01)



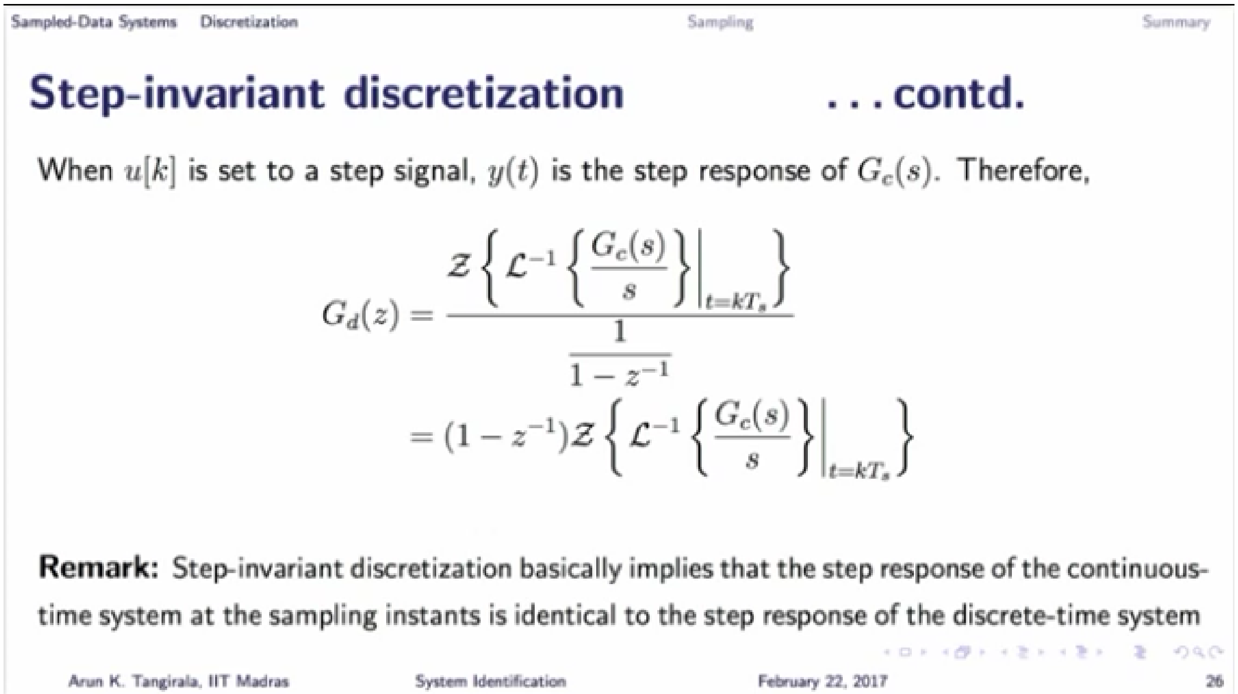
that is the idea behind the procedure, so let me go with that procedure quickly step by step, first choose the discrete time input,

(Refer Slide Time: 09:14)



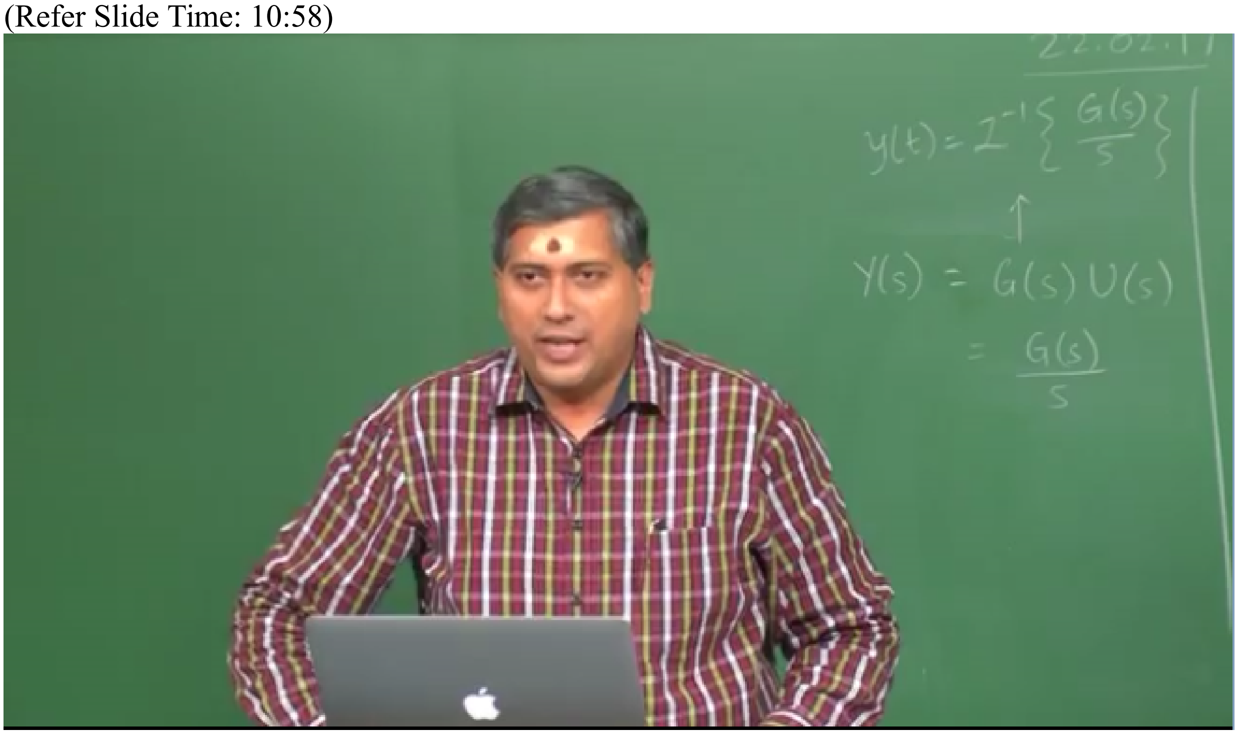
usually a step as I said you could even choose an impulse, but then you have to be careful, you have to determine the pulse response of the continuous time system, if you choose the step the advantage is, the continuous time input remains the same, step response can be calculated easily and so on, but a generic procedure is for any discrete time input that you choose determine the continuous time input that comes out of the ZOH, and 3 compute the response of the system, continuous time system to this approximated U(t), then sample it and then use this, call this sample version as Y(KTS) and then use this formula.

When you use a step it becomes easier, this procedure becomes simpler that's all, right, as a result what happens when I use a step,
(Refer Slide Time: 10:11)



we know discrete time step translates to a continuous time step, and how do I calculate the step response? Using the Laplace transform method when the signal is a, when the input is the step all I have, all I know is the Laplace transform is 1/S, therefore Y(s) is G(s)/S, right, and then Y(t) because I need to determine the response Y(t) is inverse Laplace, right, and I have to now evaluate this at sampling instance, why? Because I want to get Y(k),

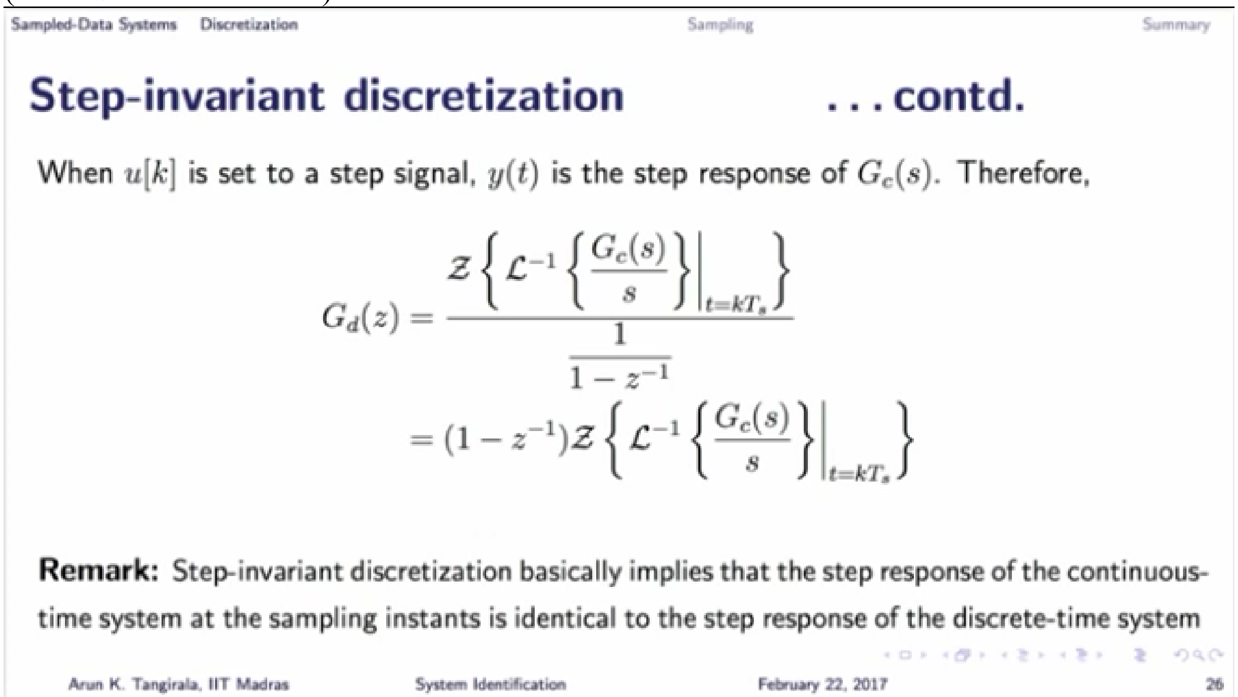
(Refer Slide Time: 10:58)



and that’s what this being done here.

So all of this now I have put it in the single formula, that's why you see so many curly braces, calligraphic symbols and everything together looks like a proper avail there, so you have here GC(s)/S is your step response in the Laplace domain, inverse Laplace gives you Y(t), evaluated at T = KTS gives you YK,

(Refer Slide Time: 11:28)

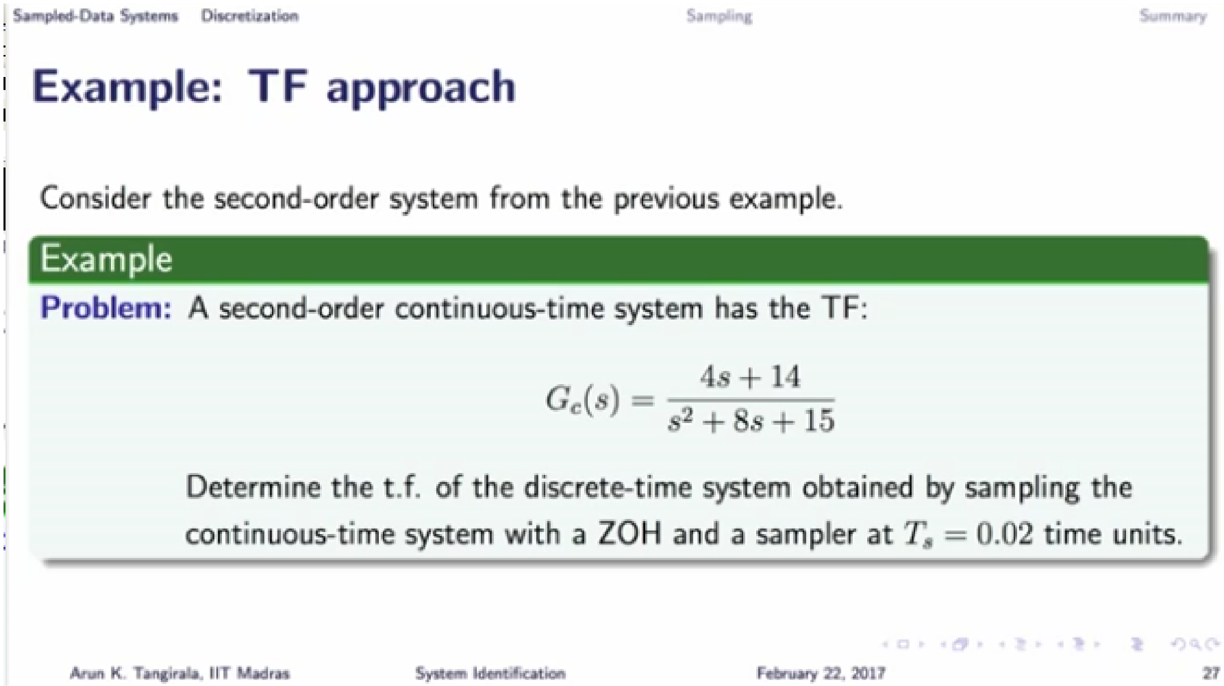


taking the Z transform of that is Y(z), alright.

And what is the denominator there? It's the Z transform of the step, and then you can just modify and this is the formula that you will find in many text books the talk of digital control, and discretization, so many a times people are left wondering how to remember this formula, it has curly braces and G(s)/S and then there is a Laplace inverse, there is Z and my God it's all confusing, you don't have to remember this formula, you just remember the concept and you will straight away be able to implement it.

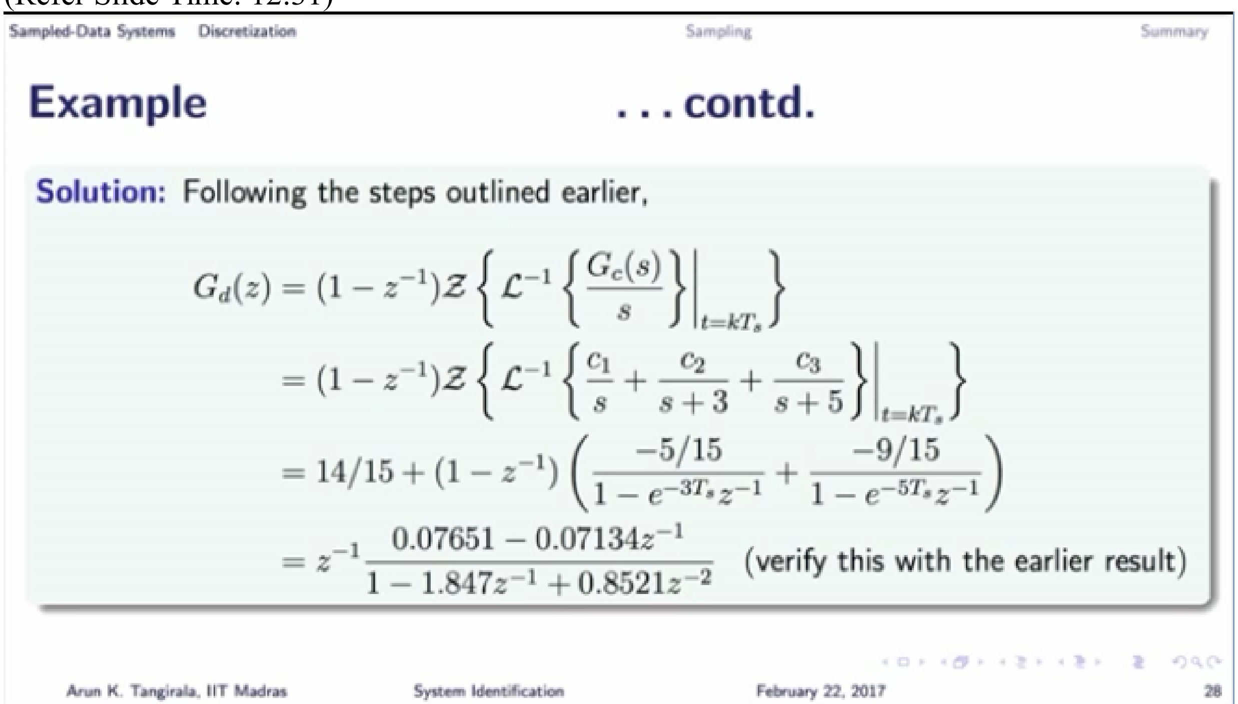
So let's look at a simple example here,

(Refer Slide Time: 12:16)



this is the continuous time process second order it has a zero as well, it’s being sampled at 0.02 time units, we have to say we are using the ZOH for discretization, straightaway we use expression that we just derived,

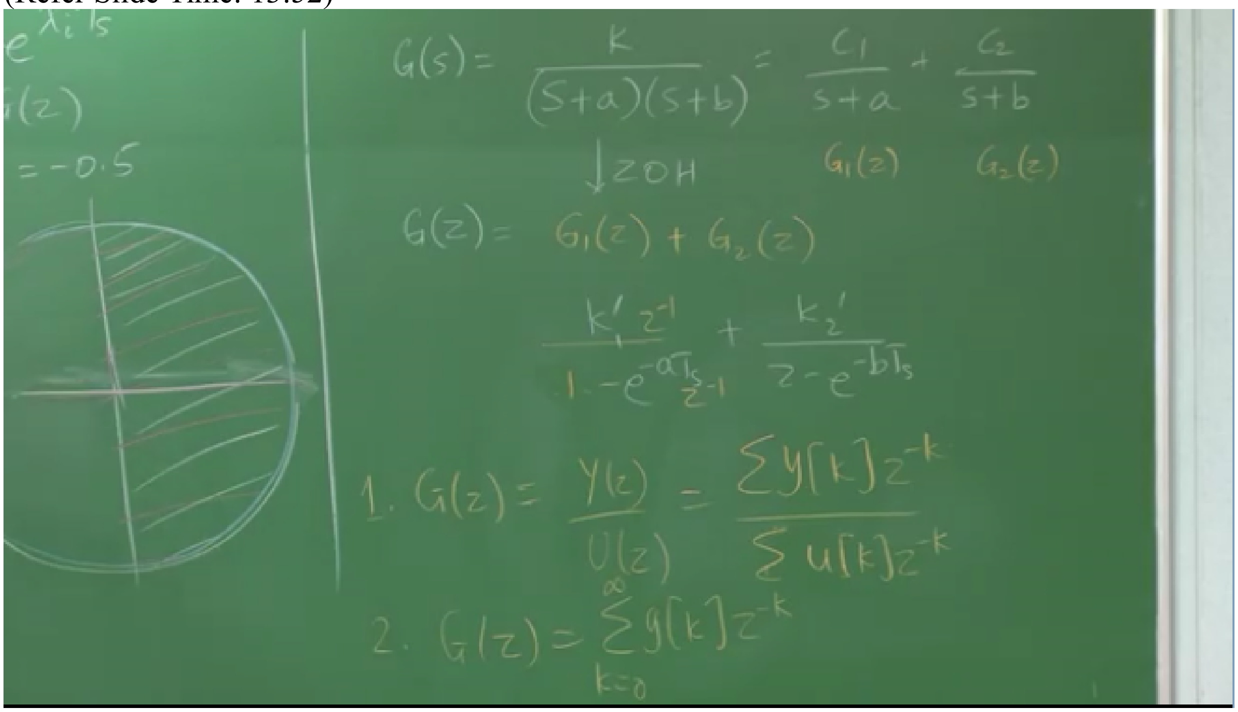
(Refer Slide Time: 12:31)



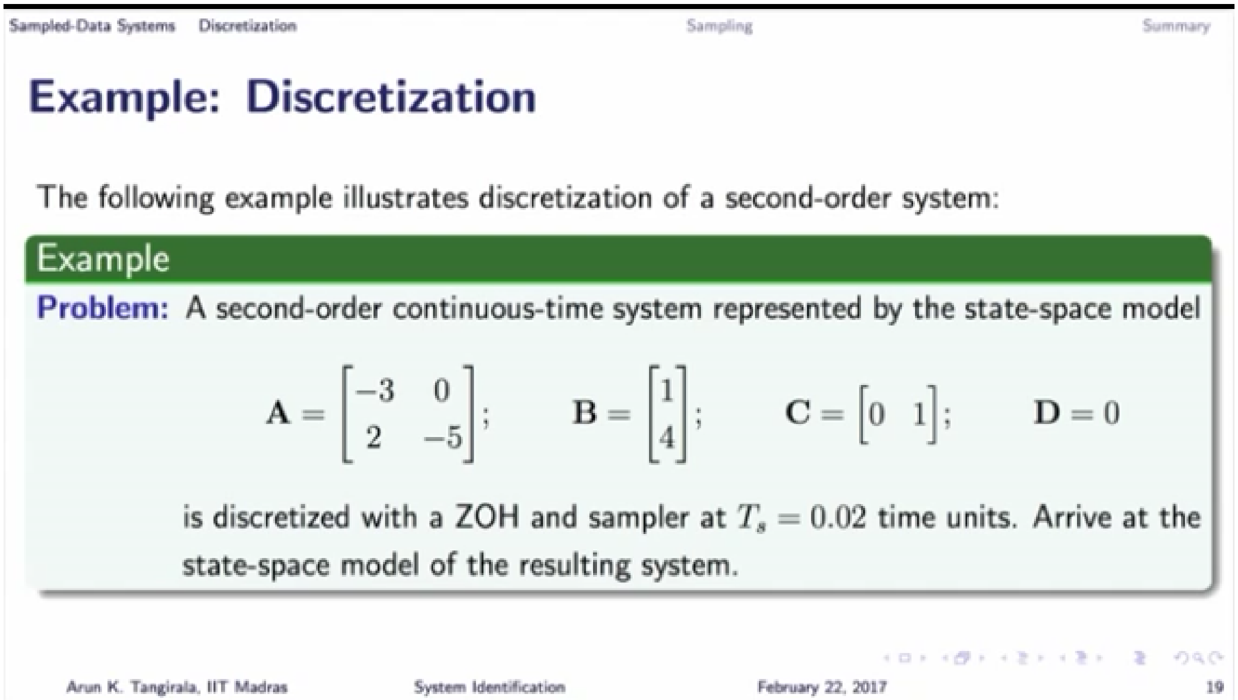
you just have to bit patient in evaluating the inverse Laplace and then taking the Z transform. And once you go through the steps here you have the discretized system, right.

Now what are the things that we noticed? Well, first that the discrete time system has a same number of polls, which makes sense we know in fact the quick check that you should, what is

the quickest check that you can do, to know if your answer is correct or not? Polls okay, anything else? Gain, gain should be preserved, right, so there are so many things that have to be correct about this and more the number of checks is here, I mean more sound verification process is polls, I had two polls, I had two polls no problem, 0 is expected regardless so that you can't use, gain preservation you can check I leave it to you to check, ZOH introduces a delay now as you can see, by the way here if you all the way I've written in this way you should see the moment you write it this way you will see the appearance of a delay.
(Refer Slide Time: 13:52)

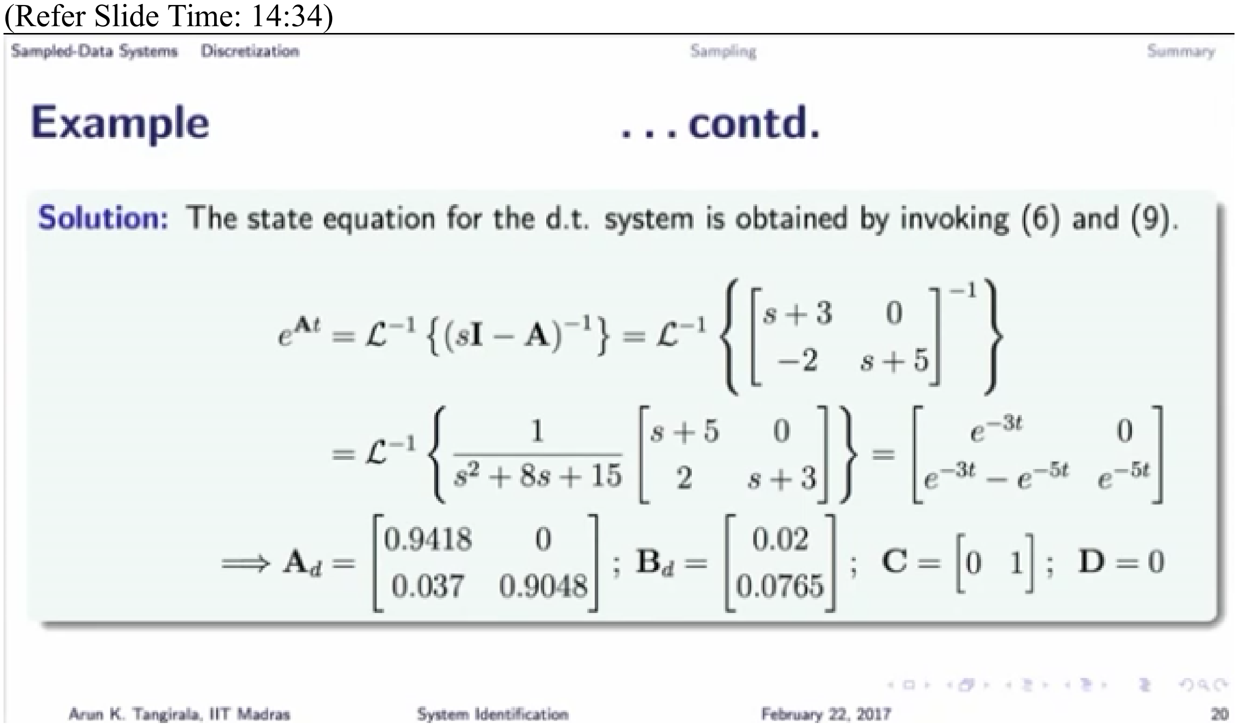


When you write K over Z – something you may not see the delay so explicitly, but the moment you write it in terms of Z inverse you can see the delay more vividly, delay appears and I'm also asking you to verify this with earlier result,
(Refer Slide Time: 14:12)



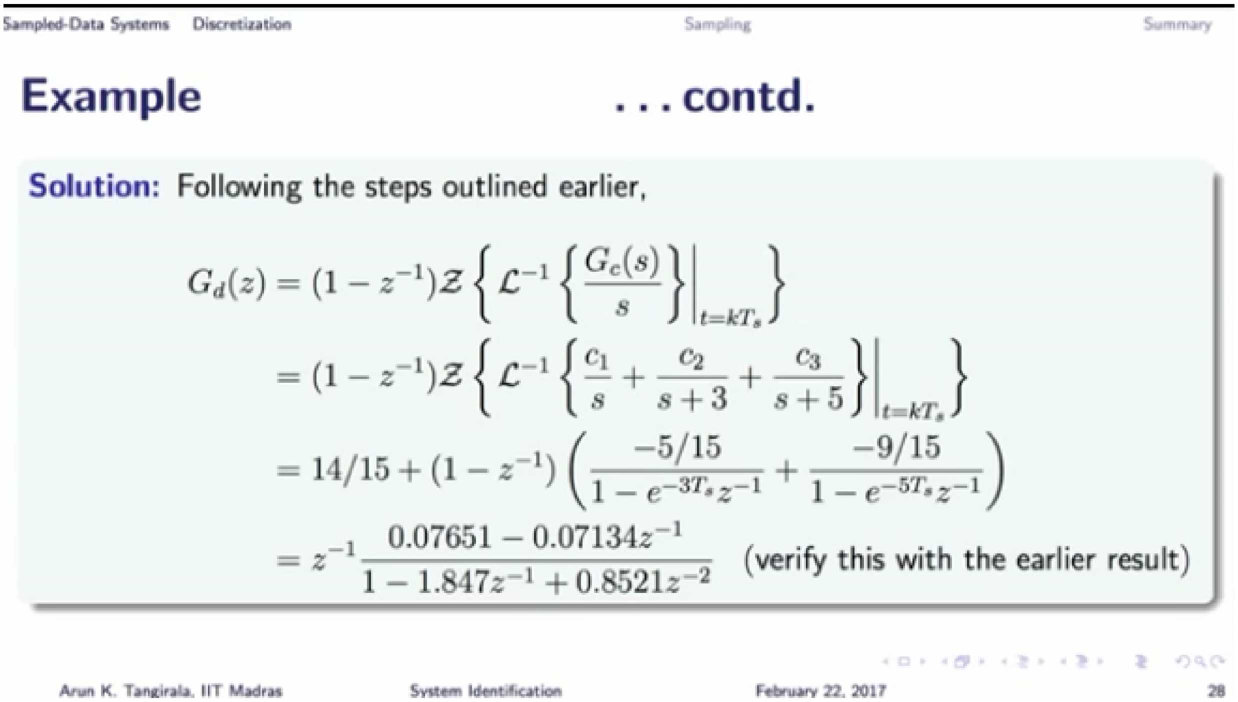
what I mean by earlier result is in fact the system here, where are the polls located? -3 and -5, and by earlier result here I'm referring to this example that we went through, it's the same system, I've written a transfer function for, and here I have state space model, so what I am asking you to do is when the slides that I'll post this afternoon,

(Refer Slide Time: 14:34)



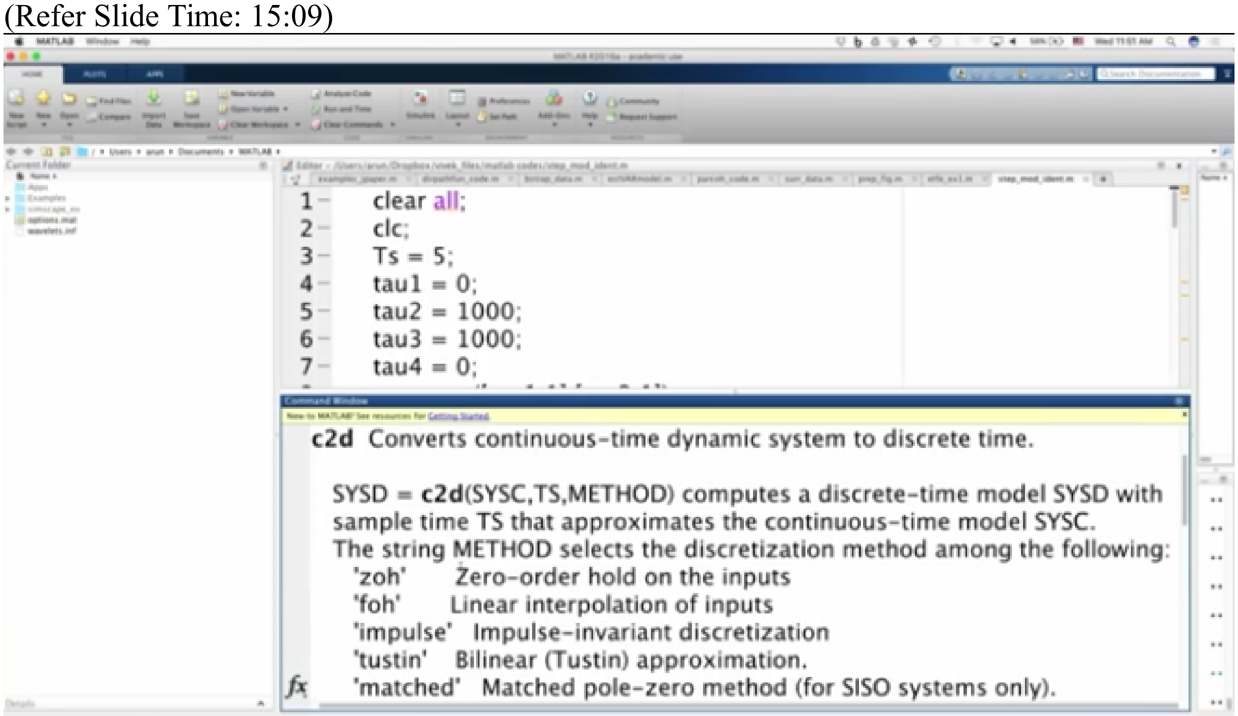
you can just check if this discretized state space model gives you the same transfer function that you have here, it should give you,

(Refer Slide Time: 14:48)



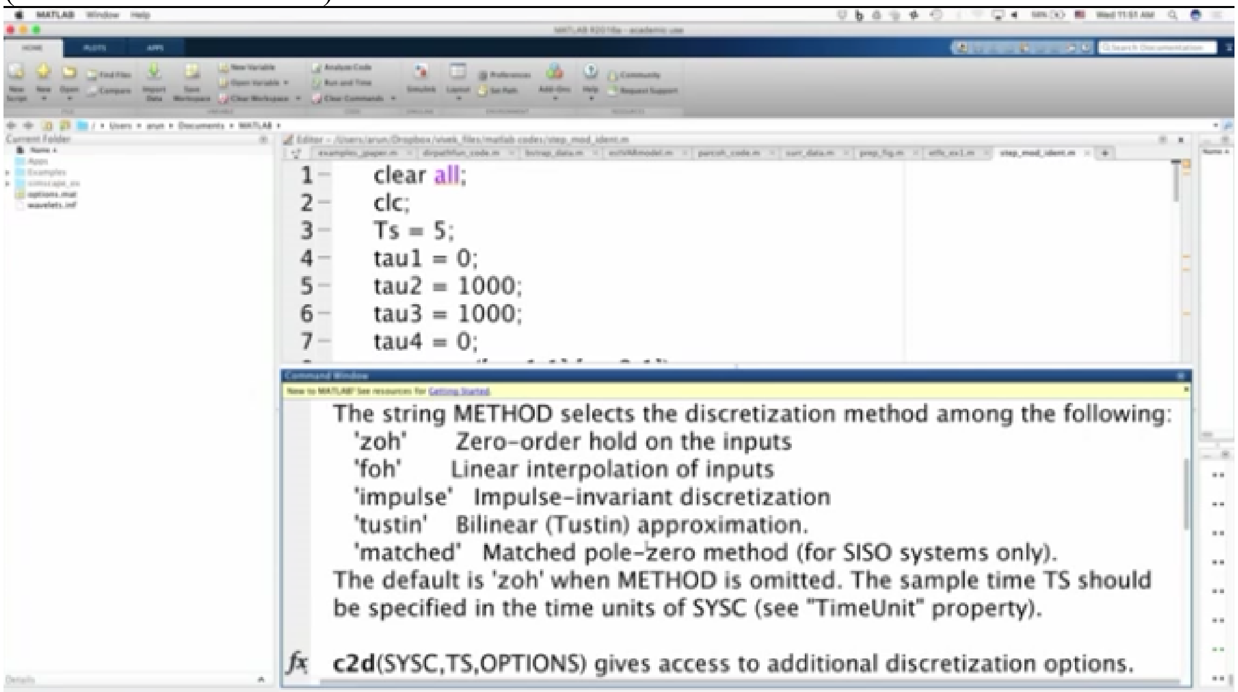
the methods maybe different but the system is the same, okay, this is what C2D does for you, right, in fact I promise to very quickly show you C2D.

So if you type help C2D you can see here,
(Refer Slide Time: 15:09)



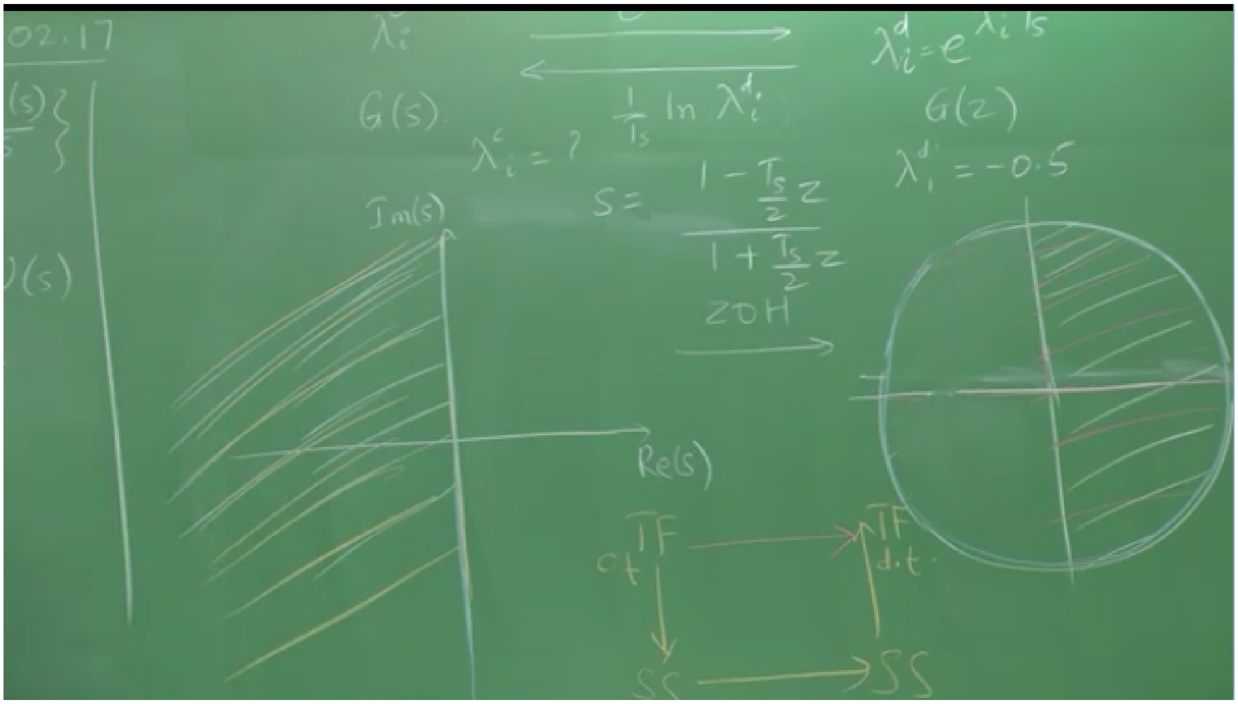
it asks for the continuous time system you can specify this as a state space or a transfer function object, TS the sampling interval obviously, method so it's asking you to specify whether you are looking at ZOH discretization which is step invariant, FOH which we don't discuss, but

we've already said it's a first order whole discretization, then there is impulse invariant discretization,
(Refer Slide Time: 15:32)



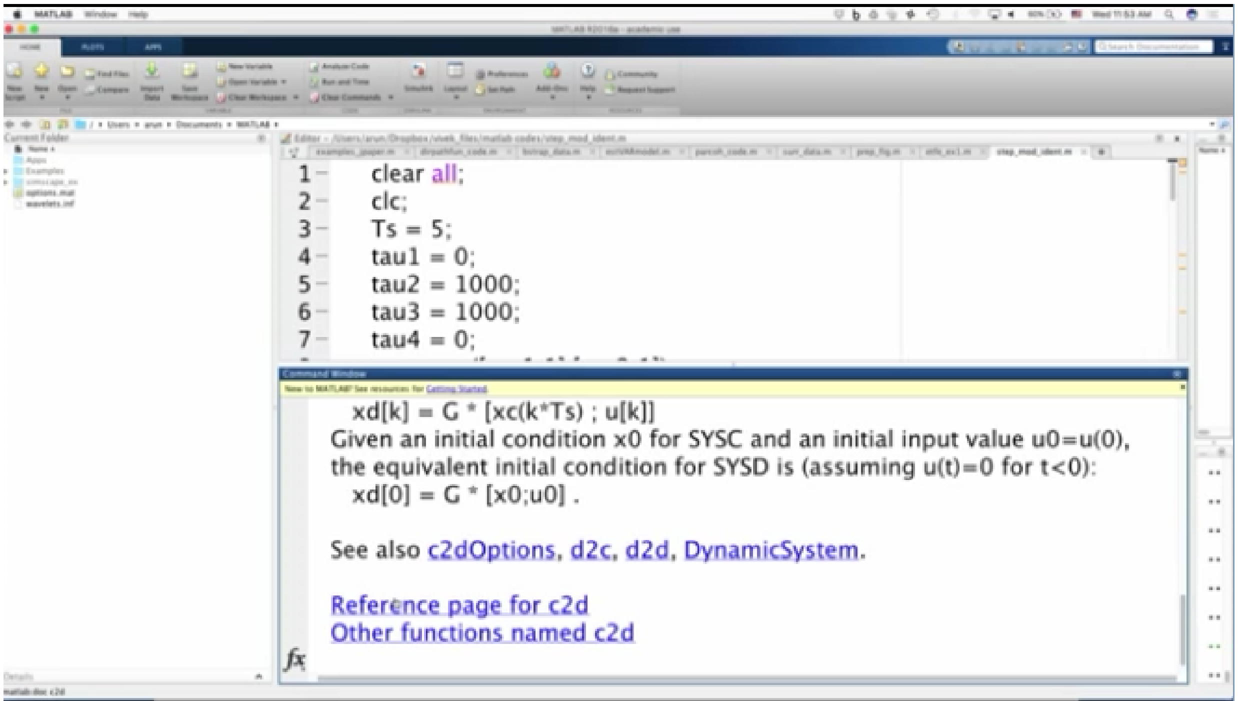
that's also possible but that no longer uses ZOH, and we don't worry about it so much but in other text books and other areas people talk about it.

Then you have Tustin approximation that I talked about, it's called bilinear, where as I said simply you replace S with a relation between S and Z, okay, and the Tustin approximation essentially says replace, wherever you see S in the transfer function go and replace that S with this,
(Refer Slide Time: 16:10)



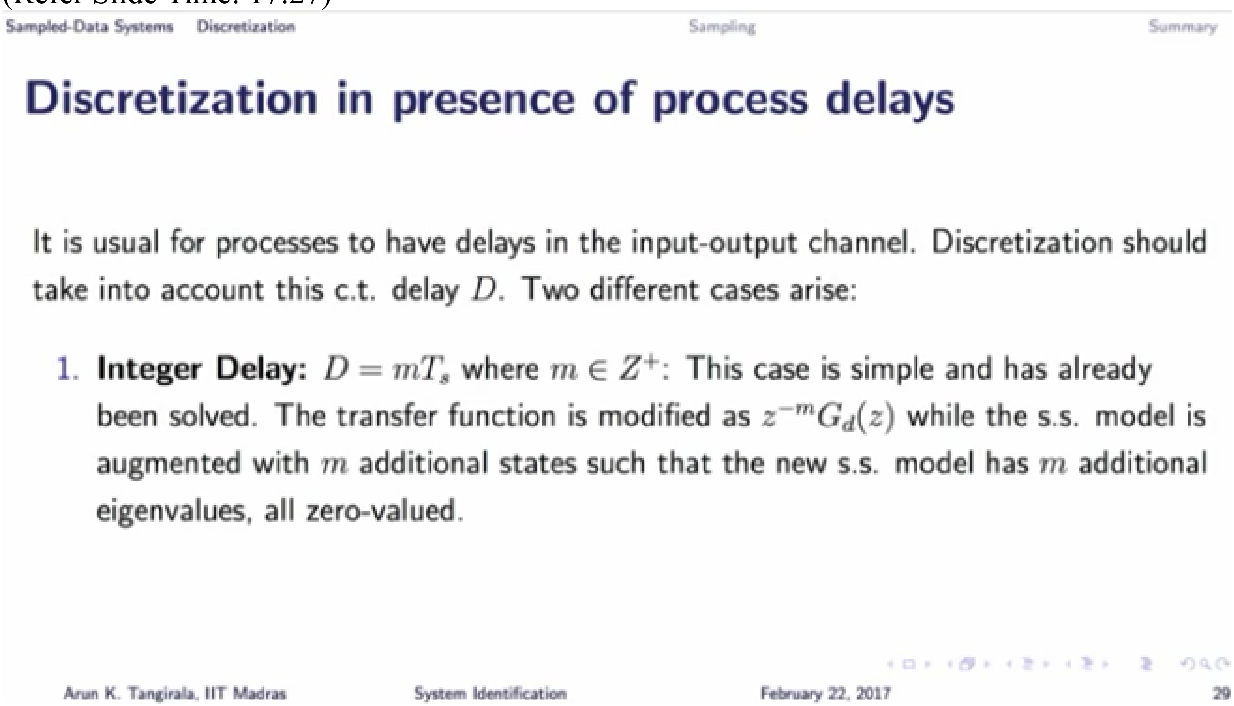
okay that is called the bilinear or Tustin's approximation, again that corresponds to some form of discretization, it's like coming up with an equivalent difference equation, then there is this matched pole 0 method which usually is done for seashore systems but again you should not use this for higher order systems, because we know 0's will not necessarily but maybe there is some formula there which is used for at high sampling rates, which says that there are 0's that are going to be introduced in the discrete time system at these locations, even though the system may not have it and so on. I've not used it extensively, but you should use it with maybe a truck of salt, okay, so generally better to avoid it.

These are the different schemes that are available methods, and it shows you what are the different procedures, its always a good practice to first read up the reference page,
(Refer Slide Time: 17:19)



because (audio gap 0:17:13 to 0:17:20) used, okay.

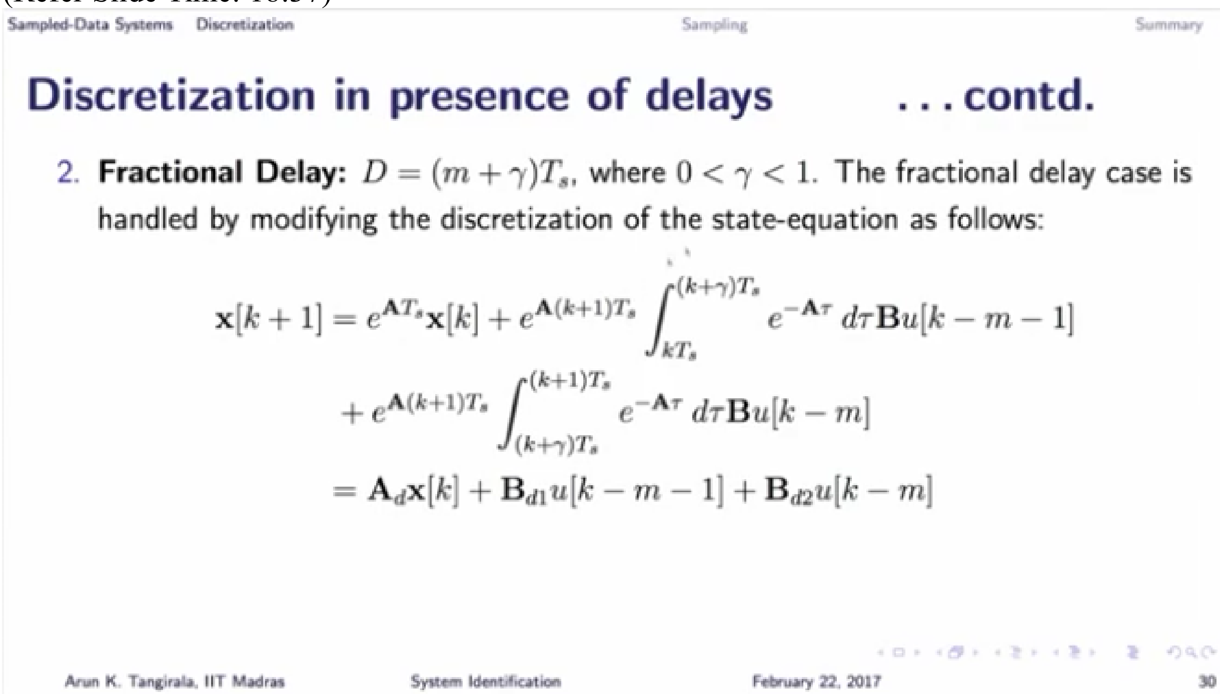
(Refer Slide Time: 17:27)



So the only thing that remains to be discussed is discretization presence of delays, no that's a very straight forward thing, I'm just going to spend maybe two minutes on it and we'll conclude.

We know that when there are delays in the system the transfer function will be modified, we've already talked about state space being modified, if the delay is an integer multiple of the sampling interval, there is no issue, you look at the delay free part of the system discretize it,

simply multiply the transfer function with Z to the –D, okay, oh sorry Z to be –M, not D here there, M is the number of integer delays that you have in terms of the sampling interval, but if you have a fractional delay that's where there is a bit more challenge involved, again here MATLAB does it for you, but you should also be aware on what is the theory behind it and if you were to drive it by hand how would you do it? You go back to your state space equation where we were discretizing, remember you recall this equation that we had, and then we said using ZOH approximation we can further simplify this integral
(Refer Slide Time: 18:37)



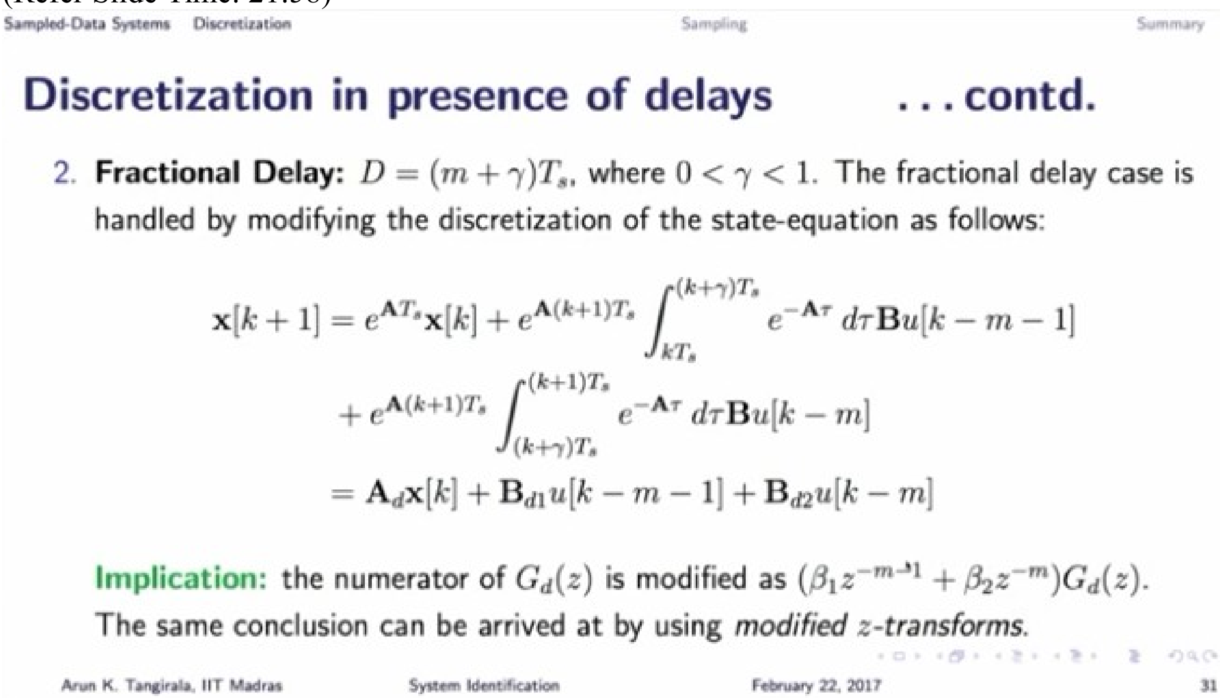
with the presence of this fractional delay, what do we mean by fractional delay? It could be 0.5 TS or 2.3 PS and so on, when you have such a situation what happens is, for a part of this sampling interval the continuous time process receives one input and the other part it receives a delayed input that means let us say the delay is 2.3 TS, then for the part of the time the input receives delayed by, the input delayed by 2 units, and for the other part of the intervals, sampling interval it receives the input delayed by 3, because it delays between 2 and 3, alright, and that is why it is important to split this integral KTS to K+1TS into two parts, one that runs from KTS to K+ this gamma is a fractional part.

During KTS to K+ gamma TS the continuous time process receives the 3 TS in our example that we just discussed, you delayed by 3TS so here if there was no delay it would have received UK, if there was an integer delay by M samples it would have simply receive K-M during entire sampling interval.

Now that there is a fractional delay from, what is this fractional gamma? That means you have M + gamma, so KTS to K+ gamma TS the process receives the previous input, K - M-1, and from K + gamma to K + 1TS it receives K – M, okay, so to do this you just have to go back and redraw the output of the ZOH and combine it with the input delay and you can visualize this fairly, but the bottom line is when you have a fractional delay the process receives 2 different inputs during the sampling interval, one delayed by the maximum which is K – M-1, the other being delayed by the other part which is K – M, as a result your state equation would have ADX(K) + BD1 U(K-M-1) + BD2 U(K-M), when gamma is 1, then you will not have an issue that means you just have only one term.

When gamma is 0 also you will have one term, it's only when gamma is between 0 and 1 you will have this two terms, correct? And you just have to evaluate this BD1 and BD2 separately. Now given the situation when you convert the state space model true to a transfer function object, what happens is in the numerator you would have necessarily Z to the -1 and Z to the –M, both, if the delay was only integer you'd only have Z to the –M, so you would have for example in the numerator

(Refer Slide Time: 21:58)



beta 1 times Z to the –M -1 + beta 2 times Z to the –M that factor will be obtained when you move from state space to transfer function that is the only thing that you have to remember, whenever you have a fractional delay the discrete time transfer function will always have this factor, which means there will be a 0 that will be introduced. Even you are looking at a first order system with a fractional delay you can have a 0.

Earlier we said first order systems and discretized will not have any 0's, but now with first order systems with fractional delays you can have a 0, you will have a 0 in fact, you can arrive at the same argument by going through the transfer functions, but then working with what are known as modified Z transforms which we don't discuss, modified Z transforms are those Z transforms that are equipped to handle signals that are shifted by fractional amount, okay, but we don't discuss that, just you have to remember that, MATLAB of course handles all of this, you can try this out, okay, so when we come back tomorrow we'll close this discussion with a discussion, brief discussion on sampling and then we will start of a review on the annum processes, alright.

Online Editing and Post Production

Karthik
Ravichandran
Mohanarangan
Wilhelm Benjamin Anand
Sribalaji
Komathi
Vignesh
Mahesh Kumar

&

Web-Studio Team

NPTEL Co-ordinators

Prof. Andrew Thangaraj
Prof. Prathap Haridoss

IIT Madras Production

Funded by
Department of Higher Education
Ministry of Human Resource Development
Government of India

www.nptel.ac.in