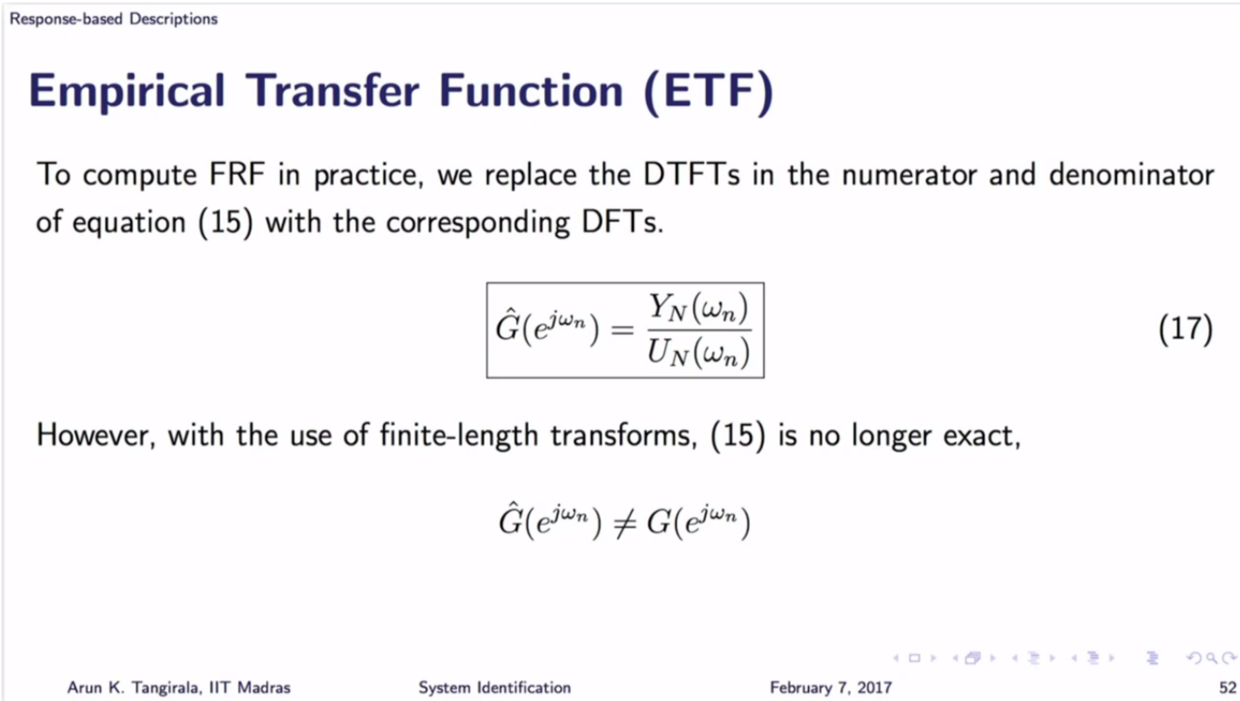
Lecture 14 Part 3

CH5230: System Identification

Response-based Description 11

(Refer Slide Time: 00:13)



So now what we are going to do is we are going to modify this. We have modified this G. We're going to-- we are, what we have done is, we have replaced the numerators with the respective DFTs, right? We have come up with a new definition of FRF. But this FRF is not going to match the theoretical one. That is whatever you see here in equation 17, have I put a hat there, if you can see carefully there is a hat, okay? And we always use hat to denote estimates. And you know the theory behind that, right? Some of my students know it.

When you wear a hat what happens? I can never know uniquely what you are? Whether you are bald, whether you are color of the head is orange or yellow or what it is, mixed colors, rainbow colors, we never know. So, the full you is never visible to me. So I only get an estimate of you. Therefore the symbol hat is used to denote estimates. Is that clear? If at in any interview somebody asks you why people use hat in estimation for the estimates, you can give this.And if they ask you who told you, whether do you find anywhere you just refer to the video lectures.

Okay. Anyway, so now why did we use hat? Because we know, that when I replace this with the DFTs I'm not recovering this G exactly. Even at that Omega [end 1:54]. By the way earlier I said, what if I use a finite spacing than 1 over n, can I get more information? No. You'll get some interpolated value, but you won't get new information. Okay. So coming back to this point here, when I replace the theoretical ones with DFTs then I get an estimate of the FRF and we call this as empirical transfer function for some reason, but it's okay, to call that.

In many domains this FRF is also known as transfer function. But you should be able to distinguish between what is a true transfer function, what is FRF? Later on you will understand the key difference between technically what is a transfer function and what is a frequency response function. But in society they are use synonymously. Okay, so we know now that this is not an exact one and that is what I've said here. The G hat of e to the J omega n is not the G at e to the j omega. It isn't? I've been writing on the board minus and there's a minus sign missing here, but don't worry about it so much, it doesn't matter.

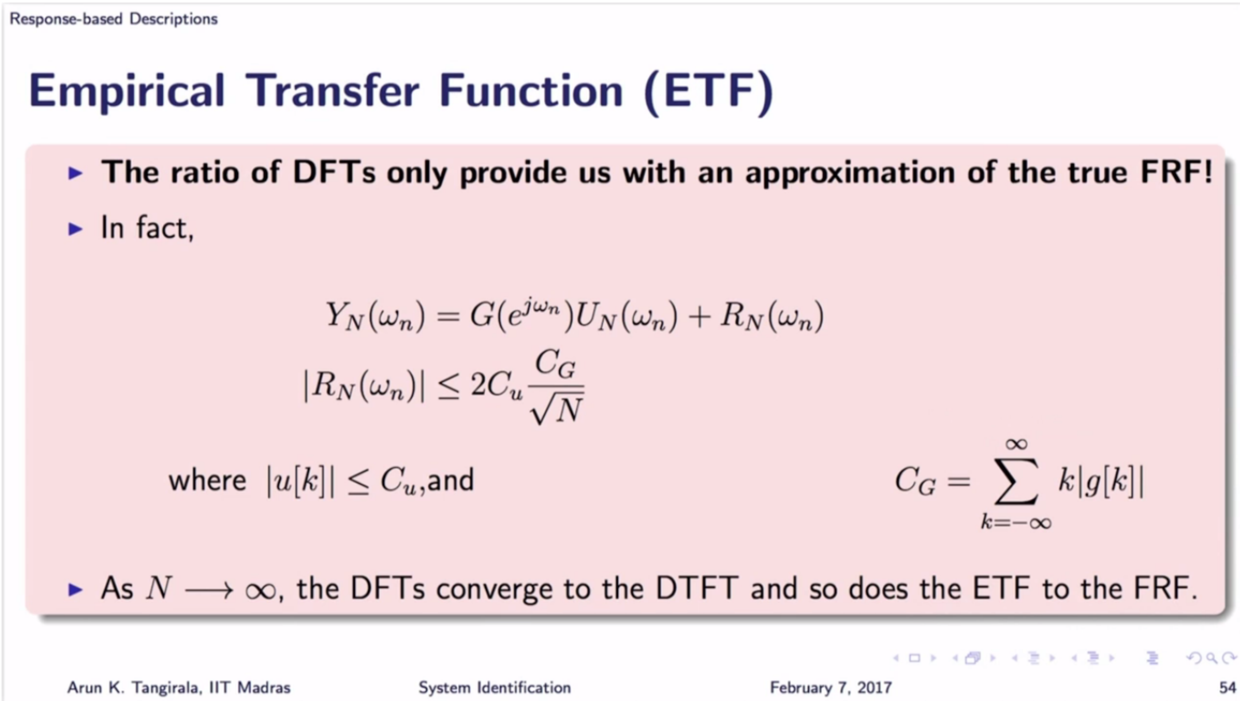
So the fact is this empirical transfer function even in the absence of noise is only an estimate. So the measurement error is yet to come. If that comes in, it get introduced an additional error. That's why I said these two alone is what we'll focus, right? So the finite data is the one that is causing the problem. Later on when we bring in measurement error, we will add another hat to this. Because that will introduce another error.

So this we call as the empirical transfer function and we can say that this empirical transfer function is exact. That is this inequality vanishes and becomes an equality, if and only if, you're dealing with periodic inputs, that means if for any arbitrary input, if you, if you compute G hat as in an equation 17, there then it won't equal to theoretical one but if I use periodic inputs, such as sinusoids or any other periodic input and the number of observations is an integer multiple of the period. So suppose I use a periodic input, let's say, its speed rate is 10 seconds. Okay, 10 seconds and I have collected, let's say 500 observations, 500 seconds worth of data with sampling interval 1 second.

Then N is 500, period is 10, so I have 50 instances of the each cycle. In such situations the G hat that I compute exactly equal to G. So that means only for periodic inputs the equality holds, for arbitrary inputs the equality doesn't hold. Why does it turn out to be that way? Maybe you can prove it. I'm not going to prove here, but you can take a periodic input write the DFT.

And then show that if the signal is periodic, then the DFT equals DTFT, they just differ by a factor, right? Obviously, if I take a DTFT of a periodic signal, in fact you can't directly apply DTFT of a periodic signal, remember DTFT assumes that the sequence vanishes at large times. So the proof is going to be a bit more involve, keeping aside the proof, remember that the empirical transfer function equals the theoretical FRF, whenever the signals are periodic with an input with the number of observations equaling the integer multiple of the cycle. So let me just quickly show you an example and then we'll conclude.

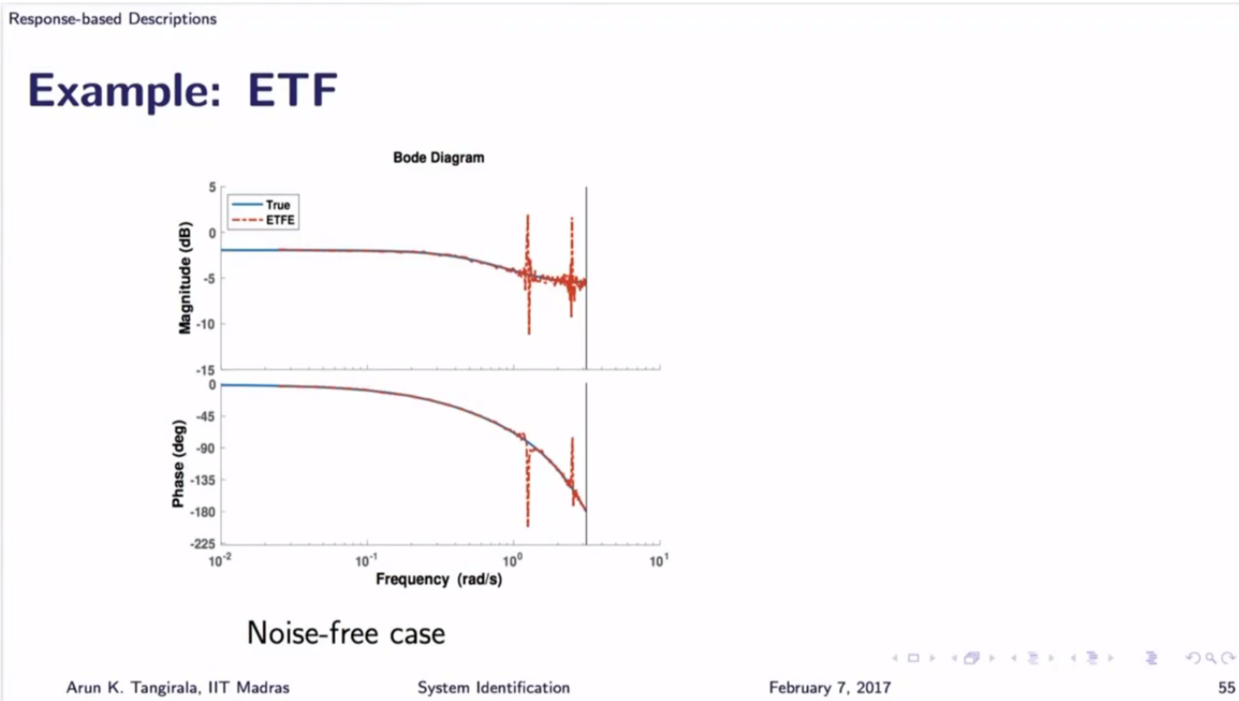
(Refer Slide Time: 05:53)



So, before I show the example some theoretical result, which talks about the difference between G hat and G and that we call as a reminder term. In other words, when I use here the DFT that is YN and UN then the relation is no longer mapped by G. So if I were to write that relation there, using DTFTs of input and output you would get Y of omega as G of e to the j omega times U of omega. This is from this definition itself straight away. But when I replace these DTFTs with DFT then unfortunately this relationship is no longer exact. There is a remainder term.

What is the nature of this reminder term? This reminder term is bounded. Is bounded above and the bound starts to shrink as N goes, N becomes larger and larger. And as N goes to infinity what happens? What happens to the remainder term? Goes to zero, right? The bound itself goes to zero, the magnitude goes to zero and therefore the remainder term vanishes. In other words as N goes to infinity, you will recover this relation, exact relation. Which means as N becomes larger and larger, the empirical transfer function becomes the FRF. So let me just show you as I said with an example here.

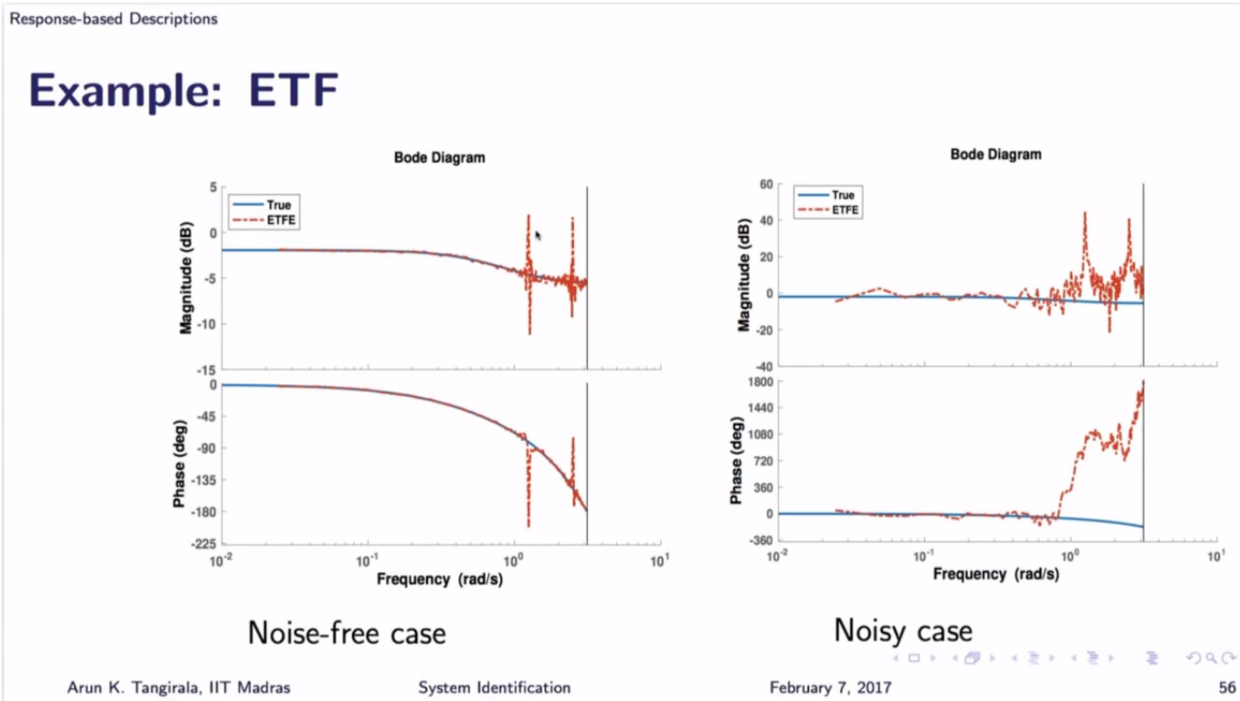
(Refer Slide Time: 07:48)



I've just taken some LTI system, don't worry about what LTI system I've taken, I've given the MATLAB script you can implement it by yourself. So what I'm showing you here is the ETF that is a G hat. And this has been obtained using the ETFE routine in MATLAB. That's why, write ETFE actually speaking it is not ETFE, I will correct the [8:16] there, it should read ETF only. We use ETFE when we work with noisy data. Because then you'll get an estimate of ETF. Right now we're only working with ETFs. So the blue line here that you see is a theoretical FRF, which I know, because I know the system.

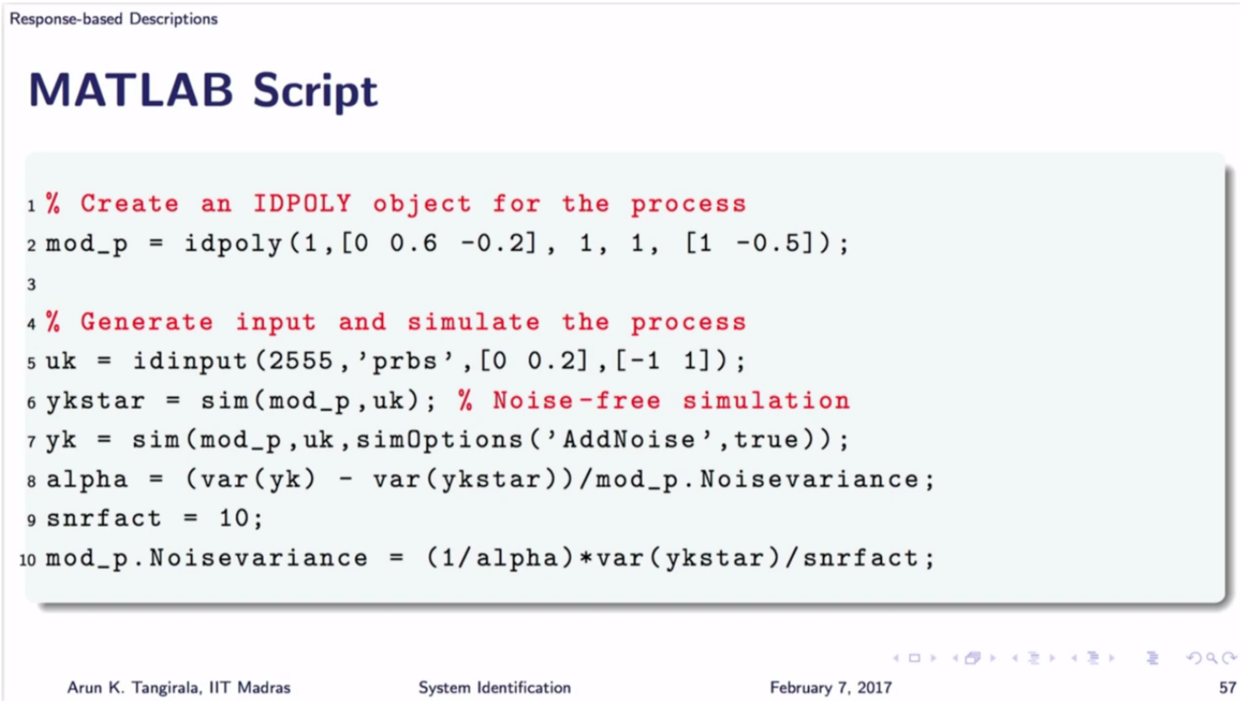
I have a mathematical description. The red one that you see in the Bode plot is the ETF. You see that there are differences between these two particularly in a high frequency regime. Of course, I've used a large number of n here. That's why at other frequencies, they match very well. You should try using smaller n, right? Where you will be asked to evaluate ETF, computer ETF and see what is the effect of n? So you can take this example, I'll pause the note and reduce the n and see if at lower frequencies also where there is a good match presently the situation worsens.

(Refer Slide Time: 09:22)



Now, when I work with noisy data the situation is evenmore bad, in fact it gets worse, even at other frequencies. Right? So this is the ETFE, truly speaking this is ETFE. What you see, because this is an estimate that I have obtained of ETF, those are G got double cap. When we derived, when we wrote G hat, we ignored this aspect. But now if you bring in measurement error you will get an estimate of ETF. And you can see the estimate is much more worse than, I mean, that is an for the noisy cases much more worse than the noise-free case. That's to be expected and nothing surprising, okay?

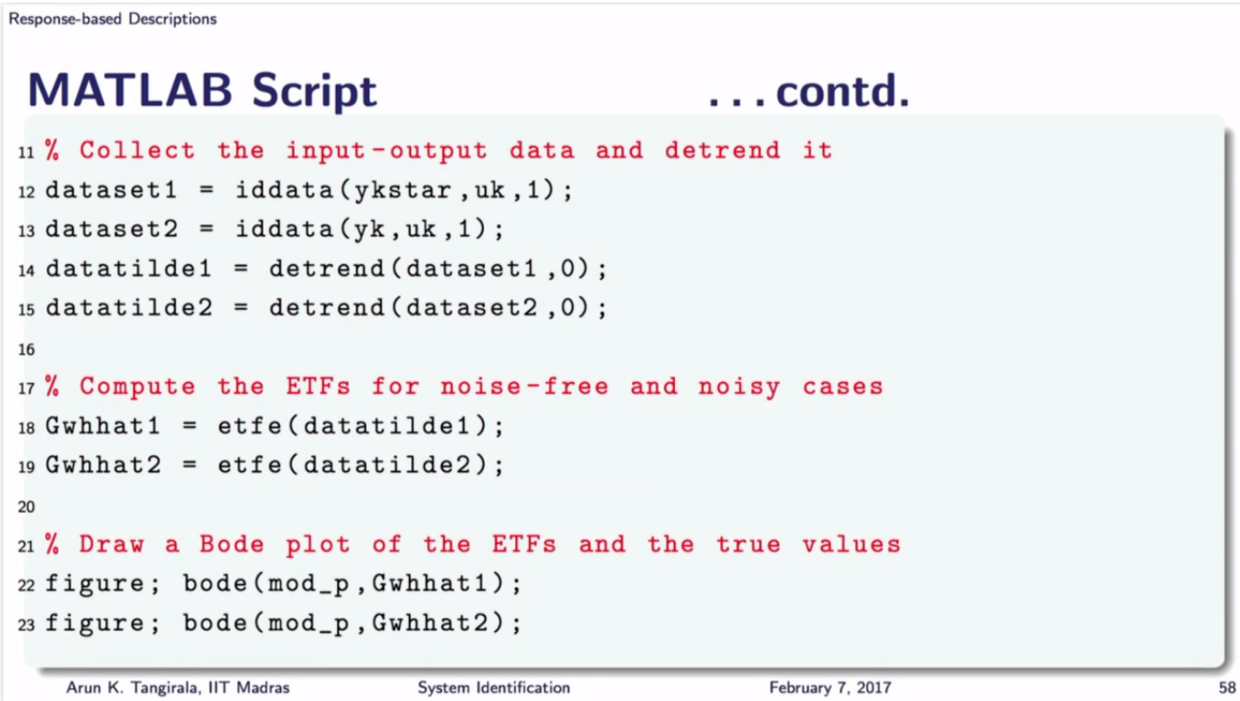
(Refer Slide Time: 10:06)



So this is the MATLAB script. I've used a second system, simple first order system. This is the idpolyis a command in system identification tool box, which takes in transfer functions. For now you can assume that I'm dealing with an output error structure. So the second polynomial is a numerator polynomial, coefficients, it has a unit delay that's why the first coefficient is zero. And this is the denominator transfer function in your G. In order for polynomial, the inputs here you have ABCDF and so on. This is called B polynomial, which is a numerator polynomial of G, this entry here.

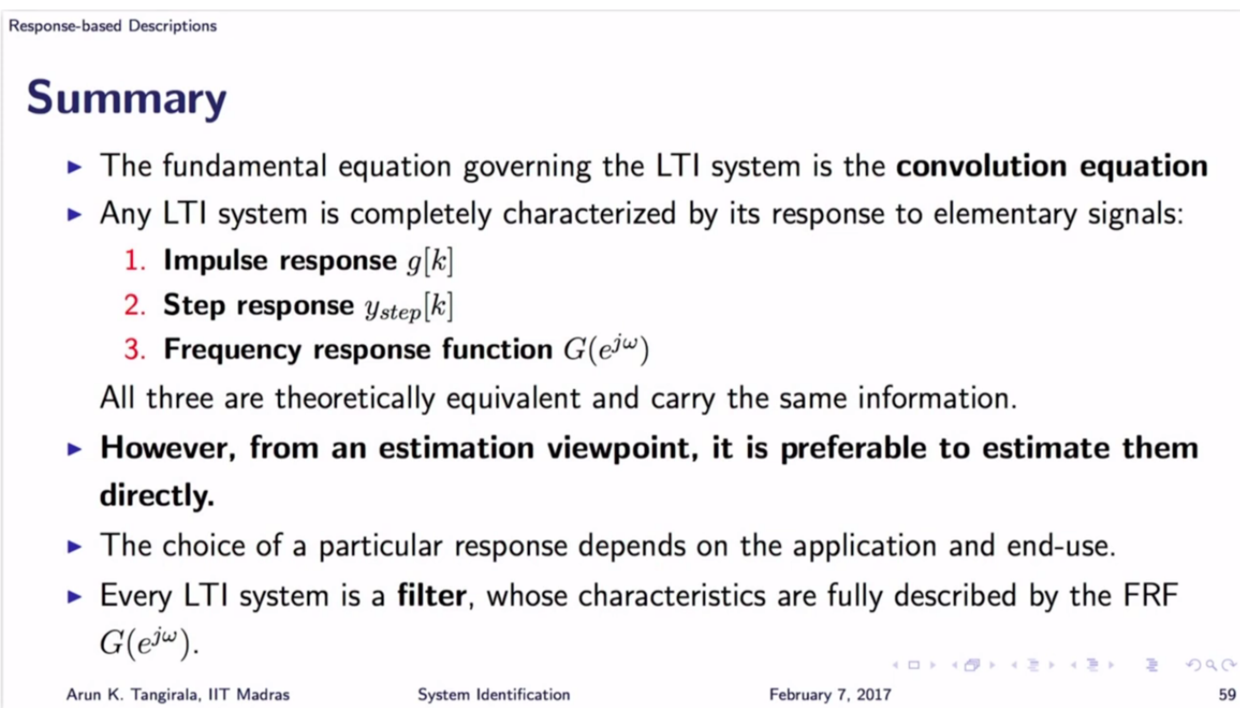
And this entry is a denominator polynomial of your G. So you can see that I've used a first order and I have generated input like in your liquid level case study and I'm using 2555 points that's too much. But that's just to show, that you will get very good estimations when n is very large. Reduce it to 255 and see whether the estimate gets worse even at those frequencies where we had good estimates.

(Refer Slide Time: 11:24)



So this is again for the noise-free case.

(Refer Slide Time: 11:28)



And run this and you'll know. So let me summarize quickly, what we have learnt in this response based descriptions, we have learned quite a lot. We have learned mainly that LTI system can be described in terms of the three elementary responses, but the mother question is always a convolution equation. And these three elementary responses are impulse, step, and frequency. And that all three are theoretically equivalent. However from an estimation viewpoint, they are different. And that we have made, we have to make some modifications to each of these descriptions, to make them identification friendly. Right? With the convolution model, we introduced a modification and we said we'll work with FIR models, very well remembering that FIR models are good approximations, depending on how large m you choose, but only for stable systems. And with FRF we introduced a modification known as a empirical transfer function. Later on, when we learn the spectral analysis part and so on, we will have to and also go through the descriptions of random processes where we review how the random, we have only focus on the deterministic part.

We have not talked about how the vk that stochastic component is described. When we complete a review of that, we will have to put them together. When we put them together we have to be careful, because this is a deterministic signal and that's a stochastic signal. You can just plug them in just like that. Like in any hardware component, you can bring in two components and put them together. You have to do it in a seamless way. At that point in time, we will learn a few more variations of whatever we have learned. And things get pretty hard there, okay? So, hopefully this gives you a good overview and also, of course, detailed theory of the descriptions of deterministic LTI systems only from a response descriptions viewpoint. Tomorrow we will look at parametric descriptions. We will look at difference equation forms. And then eventually we'll talk of states based forms followed by discretization.