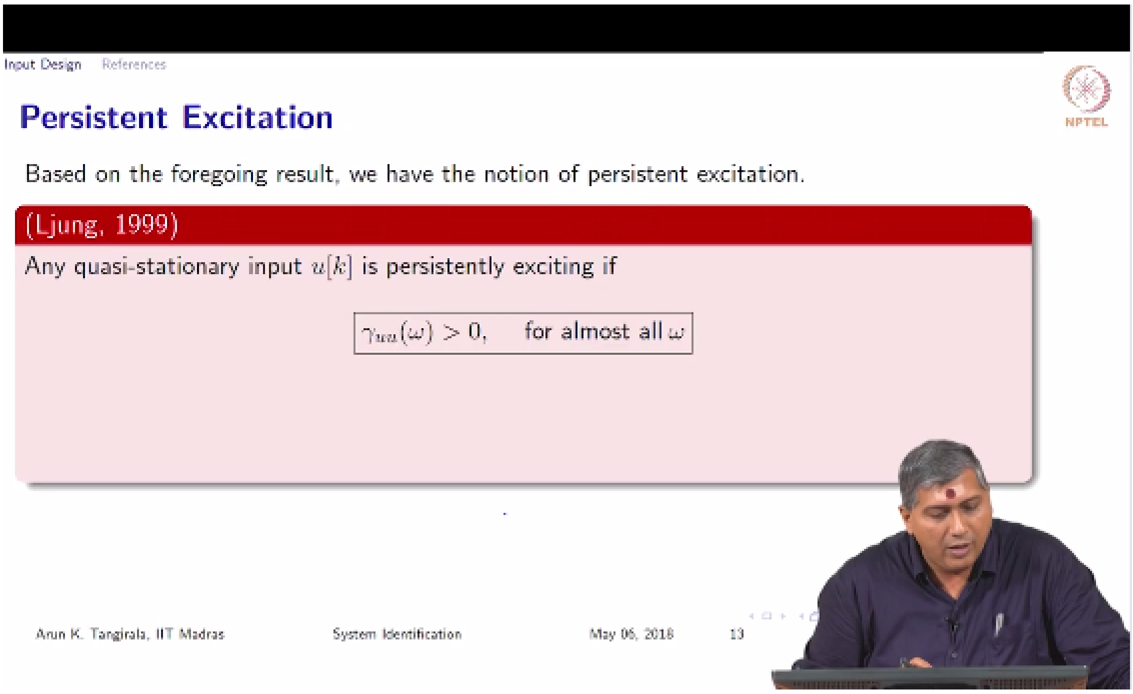
CH5230: System Identification

Input for identification

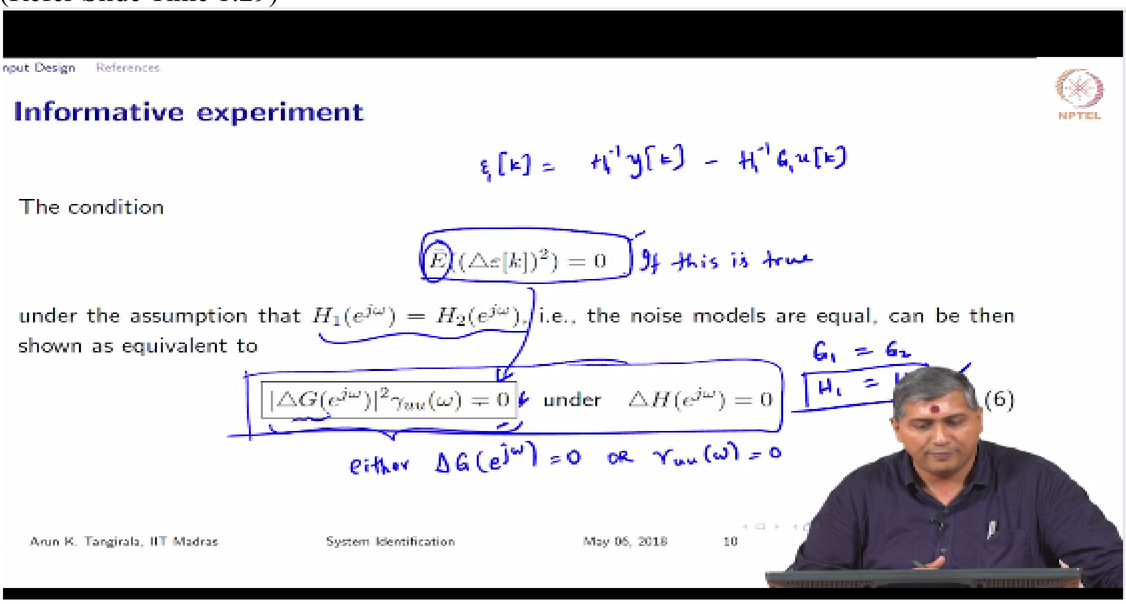
Part 2

(Refer Slide Time 00:13)



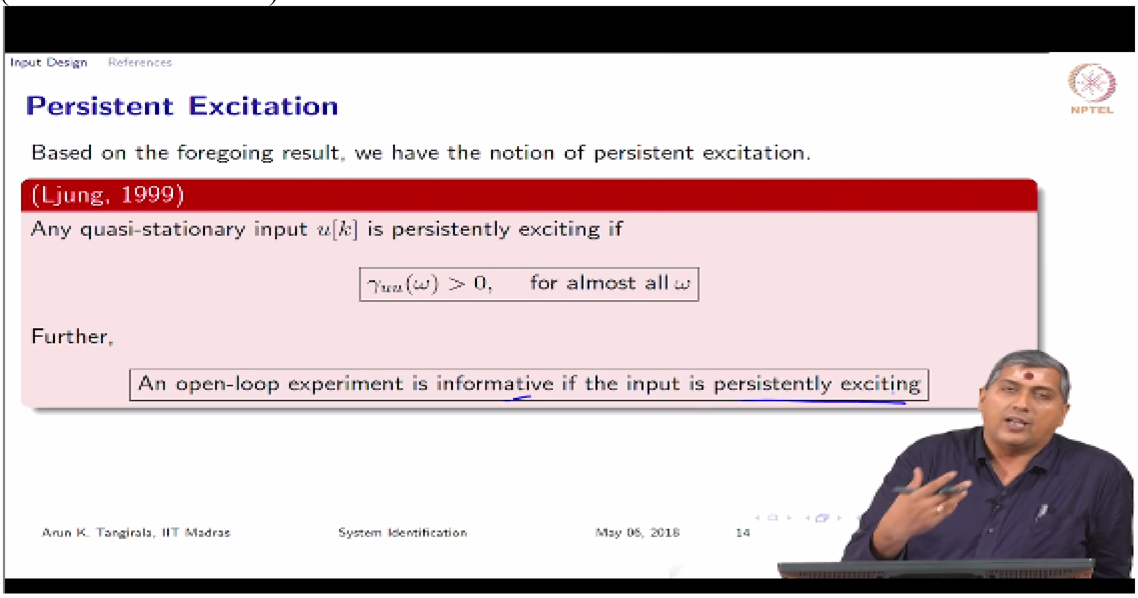
So, based on this idea, now obviously we want to be able to say that one model is better than the other not at some frequencies. That may be true for certain systems which are model systems where the interest is only in some frequencies. But for a broadband kind of a system, filters, and so on, systems that access filters, our general interest is in being able to distinguish between the suitability of two models at every frequency. And that is -- and going by the formal results that we have just had and this requirement, stems the requirement, the condition of persistent excitation. So this is what is the definition from the literature, a quasi-stationary input is set to be persistently exciting if it has power at almost, non zero power at almost all frequencies so that if this goes to zero, I know for sure that it, only that models are identical. It does doesn't go to zero, then I can use that as a measure to figure out which model is better.

(Refer Slide Time 1:29)



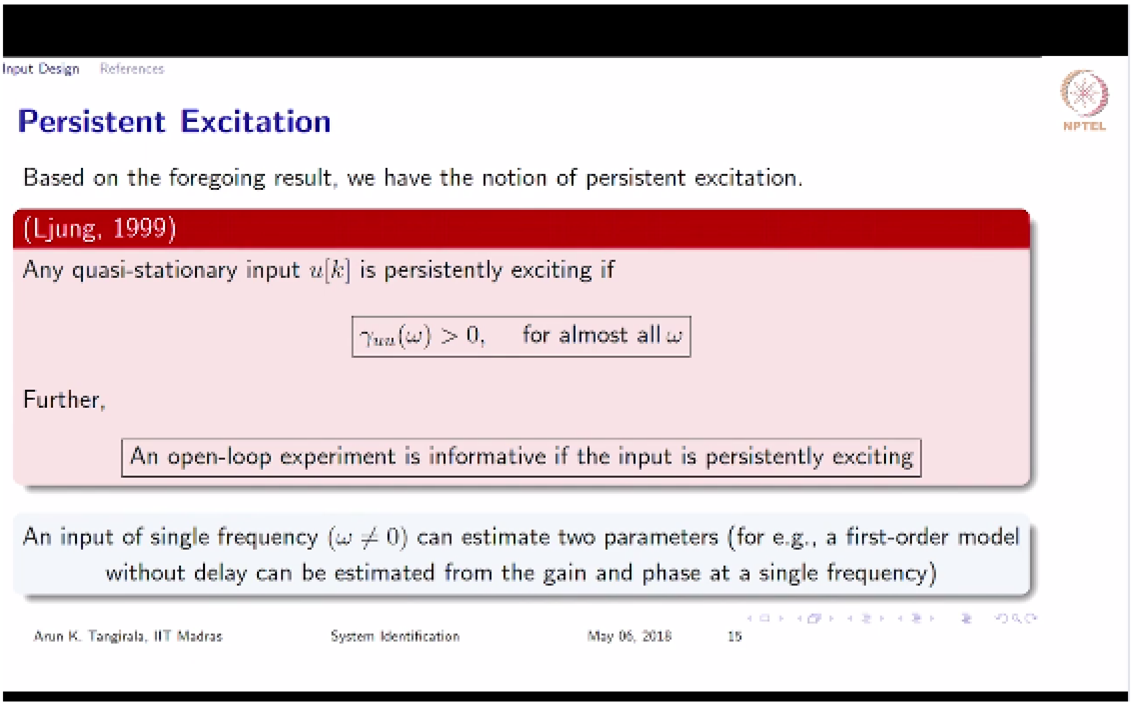
So an open loop experiment is said to be informative if the input is persistently exciting.

(Refer Slide Time 1:36)



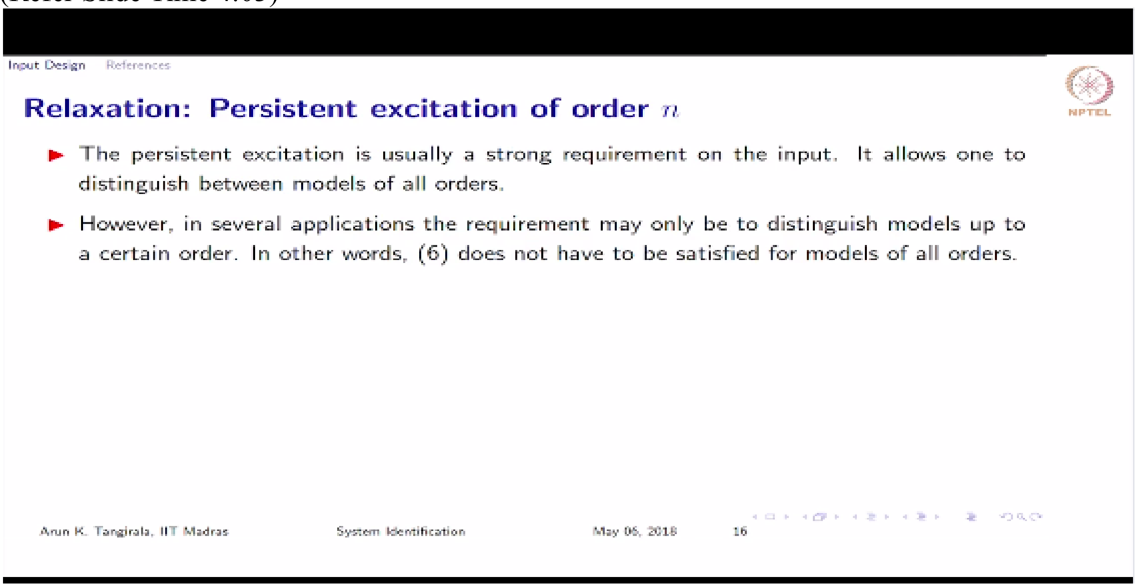
Of course, all of this is in the context of linear timing variance systems. Now, one thing to remember -- so what we have learnt is the notion of the concept of persistent excitation which essentially says, the input should contain power at almost all frequencies. A few frequencies may be excluded, okay? But at almost all frequencies. Now, is it true that I need this all the time? Yes, in general, maybe. But this seems to be a very strict requirement. A lot of inputs in practice may not satisfy this. Does this mean that I'm going to have trouble? Maybe not. Because a lot of systems do not have a full bandwidth. They have bandwidth limited to certain frequency range. And it may be sufficient for us to distinguish between, that is to decide on the suitability of two models only over the bandwidth. If I'm able to actually say that second order is better than the third order and, or something like that over the bandwidth of the system, then that's good enough. I don't need maybe answers over the entire frequency range. So to come up with a relaxed requirement, persistent excitation is a very strong requirement. White noise like signals inputs satisfy this easily. But do we want really white noise inputs all the time, right? Given that systems act as filters, they don't respond to most of the frequencies in the input. So why do I want to design a white noise input all the time?

(Refer Slide Time 3:09)



Therefore, to come up with a relaxed requirement, we first observe from our previous example, opening example, that input at a single frequency allows me to estimate two parameters. So from the board a plot if you go to continuous time domain, you know that from -- even the discrete time domain, you can understand this. If I have the magnitude and phase at a certain frequency, then I can estimate two parameters of a first-order model, the gain and time constant. But I can't do that if it is an FOPTD model. If there is a delay, then I may not be able to estimate, because there is not enough equations that I may need a second frequency. So the simple story is that input at a single frequency allows me to estimate two parameters.

(Refer Slide Time 4:03)

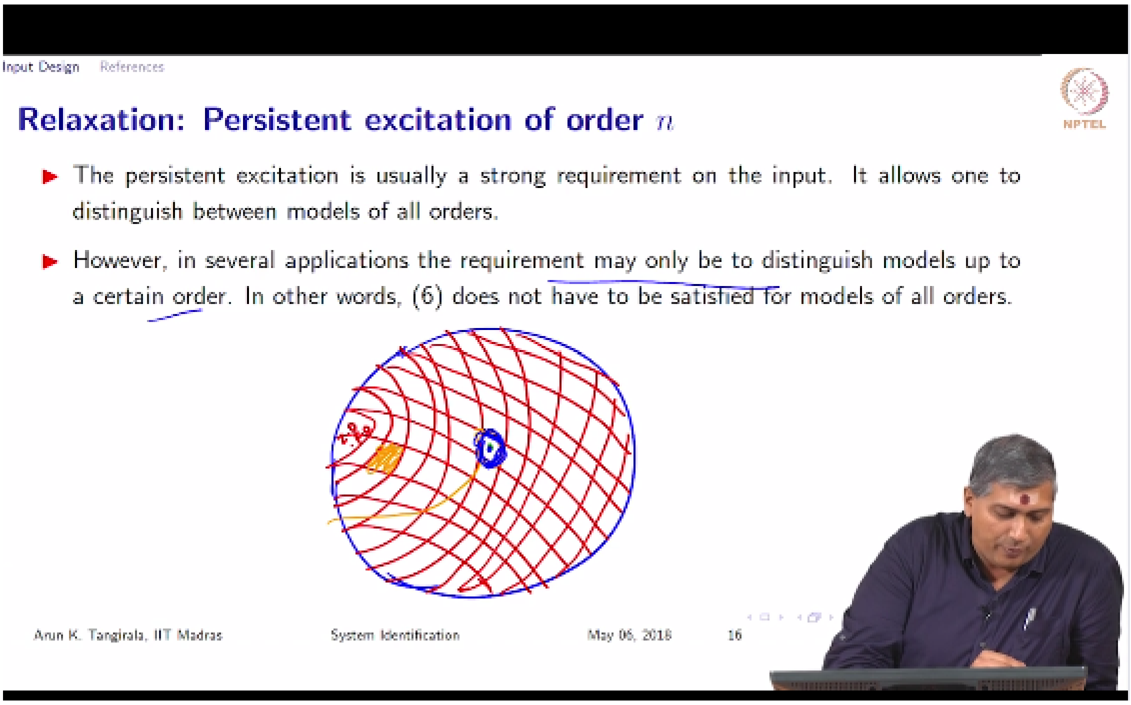


So going by this, we may come up with a relaxed requirement where, in many applications that requirement maybe only that we should be distinguishing models up to a certain order. I don't have to say whether hundredth model, hundredth-order model is better suited than ninety-ninth-order. So you see, as I want more and more ability, distinguishing ability, I need to put in more frequencies. Let me explain that to you. So imagine that your model set is like a pipe. Okay. It's not such a great drawing, but it's okay. Now let us say that you know, this is your birthday cake and now you want to partition it, right? So with a single frequency, you're going to partition it, let us say with the first cut, you will probably able to cut it this way. So with the first cut maybe I'm going to just take a small cut. Let us say this is how I'm going to cut at the single frequency. So the single frequency, remember, can uniquely estimate up to first-order. So this is the class of zeroth-order and first-order, and the rest of the pipe. So I'm now partitioned the model set into two parts with a single frequency. Then with another frequency, I will be able to estimate up to three parameters. Let us leave out the delay for now. So if there is no delay, then I can estimate up to third-order, because gain plus three time constants. So this is the second cut. Okay? So I made a first cut, second cut, now I have a final nice better partitioning. And now continue this process. Right?

So, I'm going to now with every frequency, I'm slicing it. And in fact, maybe depending on how you slice, it can also be like this because it's just a qualitative. So it could be like this also. So eventually, I want to be able to partition the model set into some very fine partitioning. This is how I want to partition. And let's say the truth is here. Okay? So let me actually use a different colour for the truth. So the truth is here. Truth is a point in our analysis. This is a beautiful diagram that helps us understand identifiability. First of all, there should be only one truth. There cannot be too many green dots. There is only one cherry sitting there and I want to pick that. Many, very often children say that, "Oh, give me the piece that has a cherry on top of it." So imagine that there's a green cherry sitting there and there is only one such piece. If there are many, then that means the model itself is not unique. Okay? So the first thing is taken care of. Second is input. I have asked enough questions. That means, I made enough cuts in the cake so that now I am in a position to isolate the piece that has the truth. But then I also need to have the ability to pick that piece which has a truth. What if I make a mistake and I pick let us say, this piece here, I missed out the truth. How many ever attempts I make, I miss out the truth. That has got to do with my ability to estimate. So it is -- now the estimators turn to be able to guide you. It should say, go here, go here and pick this one. This is the one that you want to pick right? And once you pick that, then you have, kind of, pick the one that has the truth in it. And that estimator which is consistent which will, is the one that will guide you to pick the true one. So, through a very simple example like this, we are able to understand the role of input, the uniqueness in the model, the role of input design, and the role of the estimator. And we also learn an important fact that we will never be able to pick that particular truth.

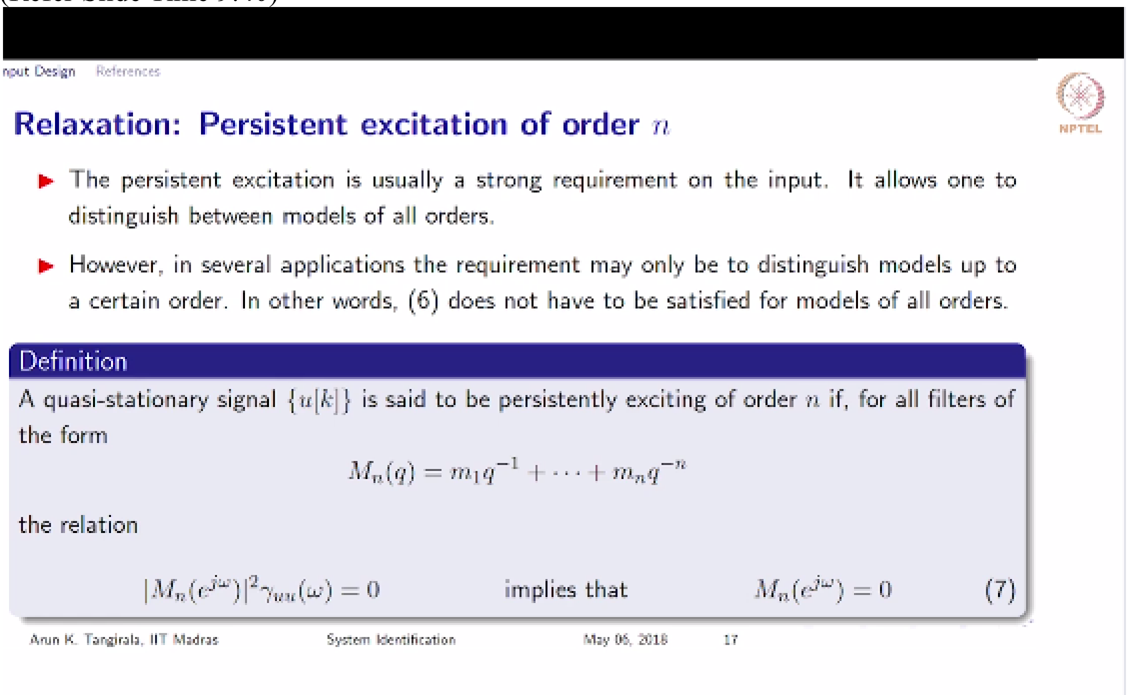
We will be able to pick only that smallest piece of pie that contains the truth, which means that we will never be able to get to the truth but we will be able to get to the interval that contains the truth. So this is the confidence region for us. Okay? Of course, in this case, it's pretty easy. I know for sure the truth is in that interval that I've captured it. In identification, we are not completely confident. There is a 95% confidence, 99% confidence and so on. Okay? Maybe when you were picking, the cherry fell into the other one. So anyway, so that's the point here. And hopefully this illustration really helped you.

(Refer Slide Time 9:20)



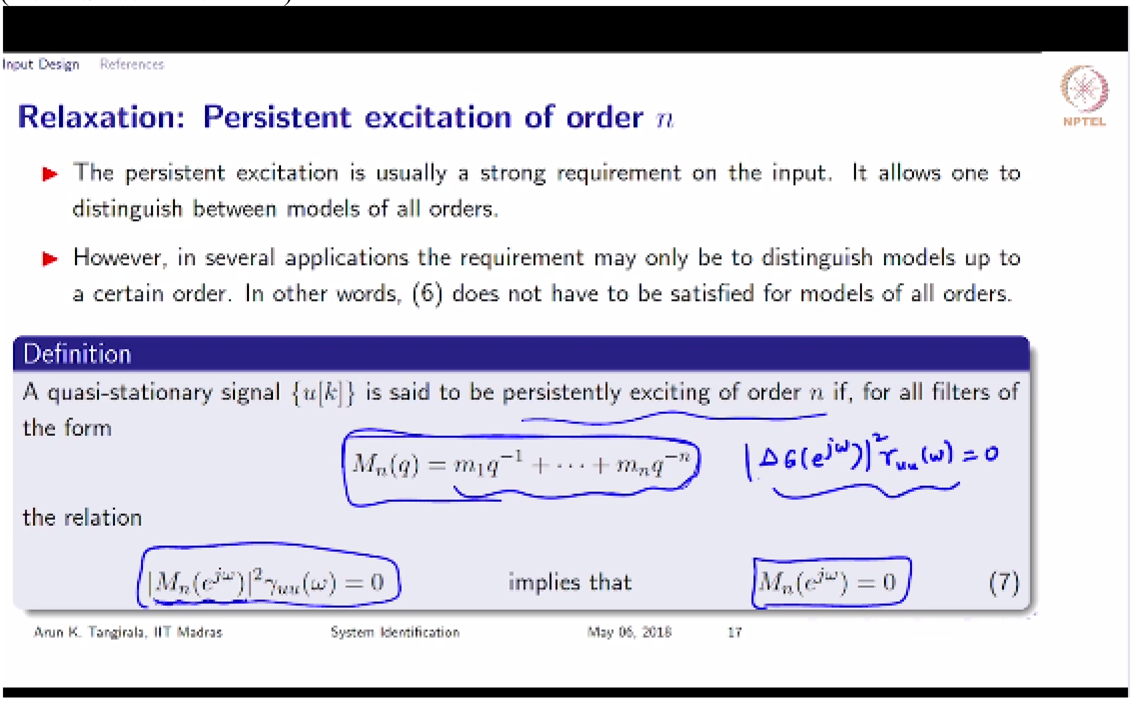
What we are talking about here is how fine a partitioning we want to have. In some cases, I may not want to a fine partitioning. And that is what we mean by persistent excitation of order N. Persistent excitation means I want to a very fine positioning of the cake, of the pie.

(Refer Slide Time 9:40)



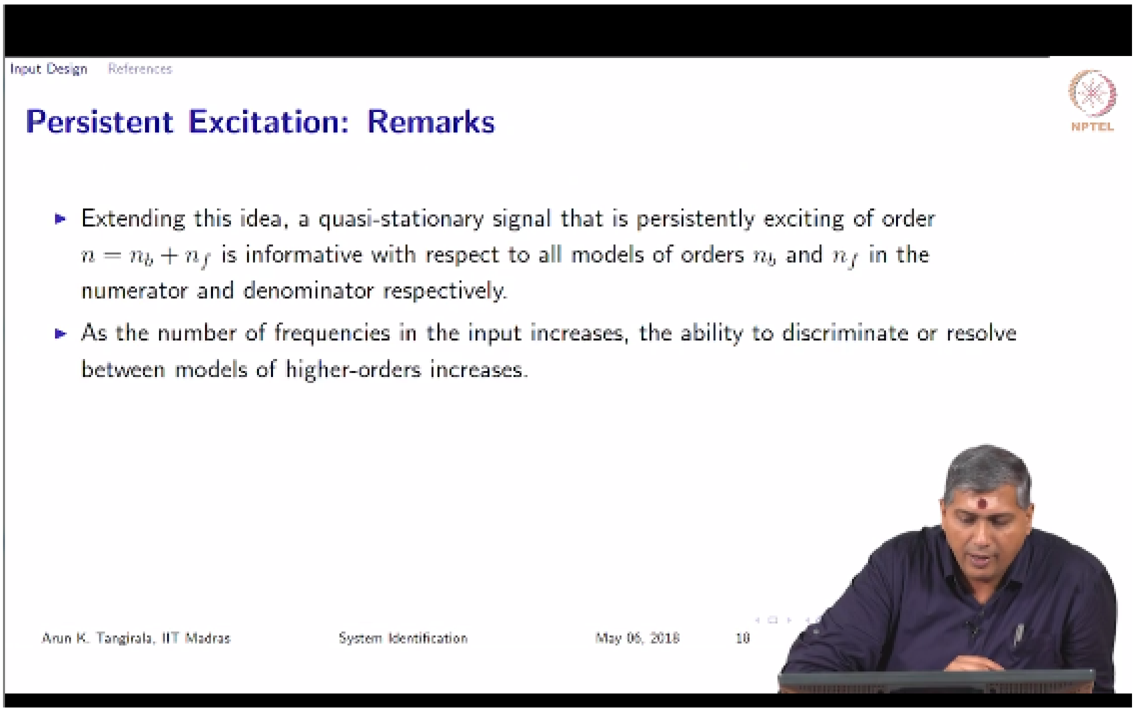
Persistent excitation of order N means, I don't want a very fine partitioning, a course partitioning is okay. And technically what this means is, it is sufficient if I'm able to distinguish up to tenth-order. For the application that I have, it's enough because I know already that the process cannot be beyond tenth-order for some other reason. For this reason, we define a persistent excitation of order N. What does it mean of order N? Going back to the earlier equation, we had delta G[e] to the j omega modulus square, gamma Uu of omega equals 0. My input -- earlier we said my input should be such that this expression should go to 0 only when delta G is 0. Now we are saying this expression need to go to 0 only when delta G is 0 up to nth order. If I choose models of higher order, it's okay. This expression can go to 0 even when delta G is not equal to 0. So, that is the other way of saying. So, now we are saying delta G up to Mn, that is, if -- up to this form is okay. When delta G has an order greater this then I'm not so bothered. But certainly I'm demanding when the delta G is are of this form, are of this order difference, then my input should drive this expression to 0 only when this is 0. The other way of looking at it is that a persistent excitation, input of persistent excitation of order n can be generated using this equation. This is actually can be viewed as a generating equation for the input as well. Either way you look at it. This is kind of giving you a design equation for the input. It says that, if you want to design an input of a certain order excitation, persistent excitation, then essentially let it be the -- let it satisfy this equation. So this is also used sometimes in the design of inputs that are persistently exciting of order n.

(Refer Slide Time 12:00)



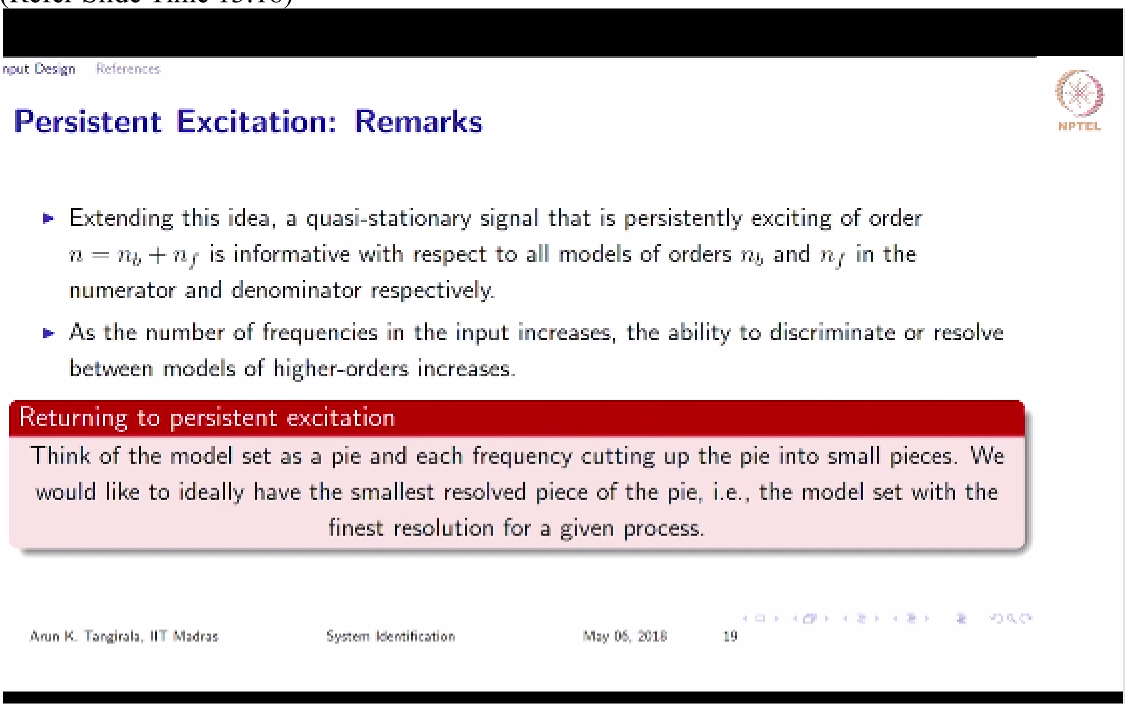
But if you don't understand any of this, all we're saying is persistent excitation is a strong requirement. Persistent excitation of order n is a weaker requirement, which allows you to distinguish between two models up to order m.

(Refer Slide Time 12:17)



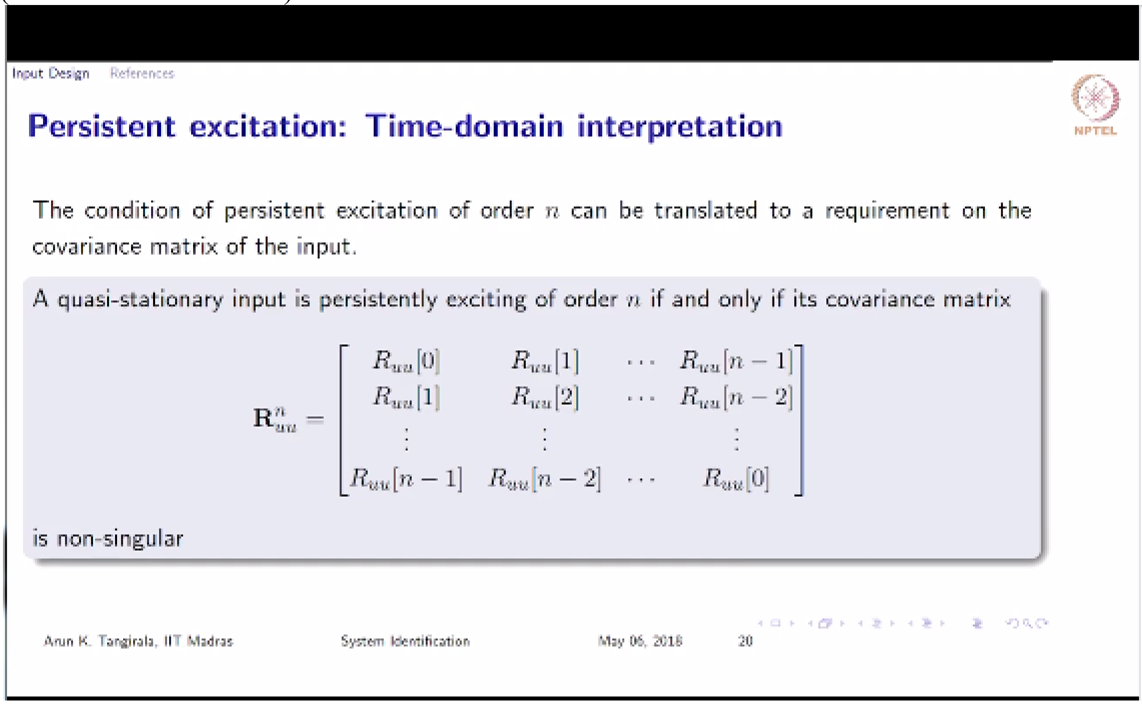
And you can now extend this to persistent excitation of order nb plus nf because if you're fitting a model of B over OE model let us say B over F, then B will have nb parameters and F will have nf parameters. So overall, I should be able to estimate nb plus nf. And we say that quasi-stationary signal that is persistently exciting of order nb plus nf will allow me to distinguish models up to orders nb and nf in the numerator and denominator. So suppose I choose nbs2 and nfs3 and if I claim that the input is informative of order five that is persistent, sorry. If I say that it is persistently exciting of order five, then I will be able to distinguish up to models which are second order in the numerator and third order in the denominator. And we've already discussed this aspect, so will not go over that.

(Refer Slide Time 13:18)



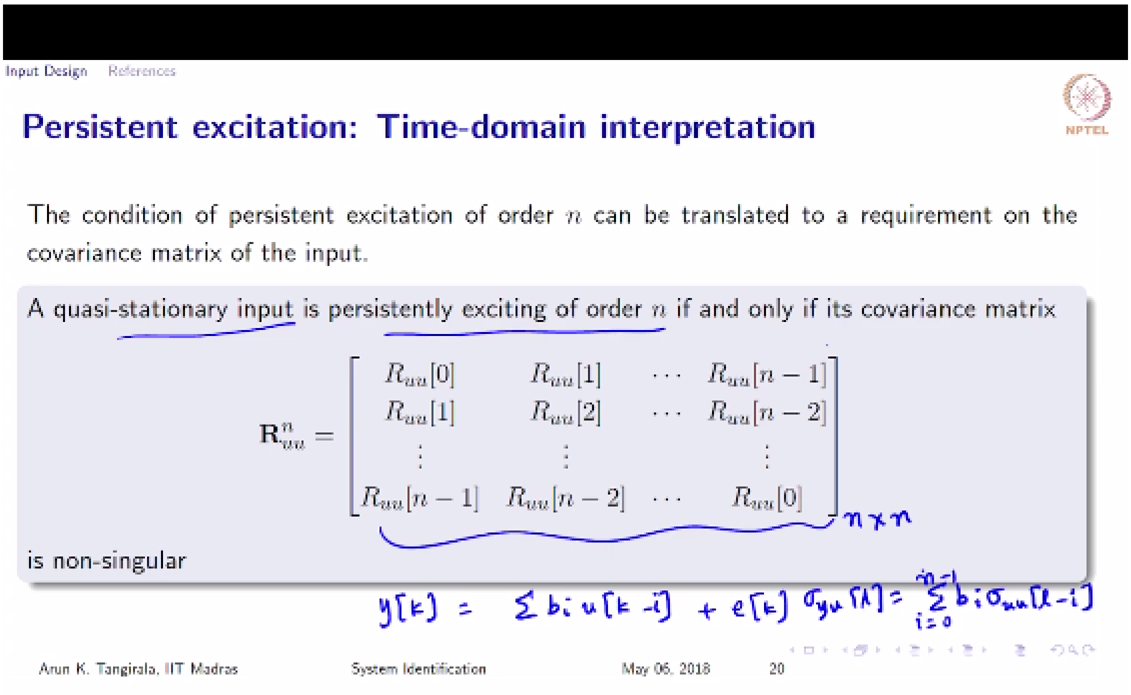
So, and returning -- so essentially, this is what I have demonstrated. So, I'm not going to go over this statement in the box.

(Refer Slide Time 13:26)



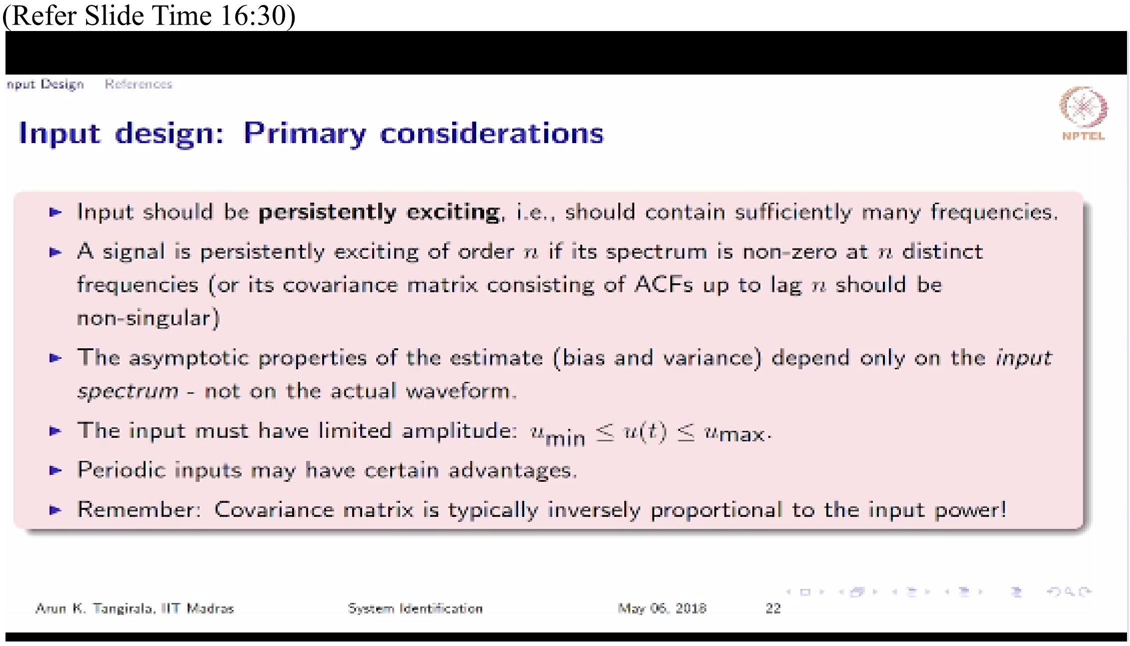
I've already, earlier I've told you that although a lot of these results are in frequency domain, there is also time domain equivalent, and in the example I did tell you that, sorry. When it comes to noisy data, we don't look at the rank of the lagged inputs. We looked at the rank of the variance covariance matrix of the inputs. So I'm avoiding a lot of derivation here just with some analogy with the previous example, and the connection -- using the connections between, sorry, the spectral density and the variance covariance matrix. In fact, the spectral densities are the Eigen values of variance covariance matrices of infinite order. So, we can use that connection and say that a quasi-stationary input is persistently exciting of order n if and only if this covariance matrix is full rank. So, this covariance matrix that we see is n by n. And again, if you again, if you're having difficulty following the theory that we just discussed, you can keep all of that aside and simply go back to your y[k] equals sigma bi u[k] minus i plus, let us assume white noise for now, and then take the covariance of y with u, and right hand side also with u so that you get sigma yu at lag l equals summation bi sigma Uu at k minus l. Sorry, l minus i. And i runs from, let us say 0 to, here in this case, n by n, so we can say n minus 1.

(Refer Slide Time 15:21)



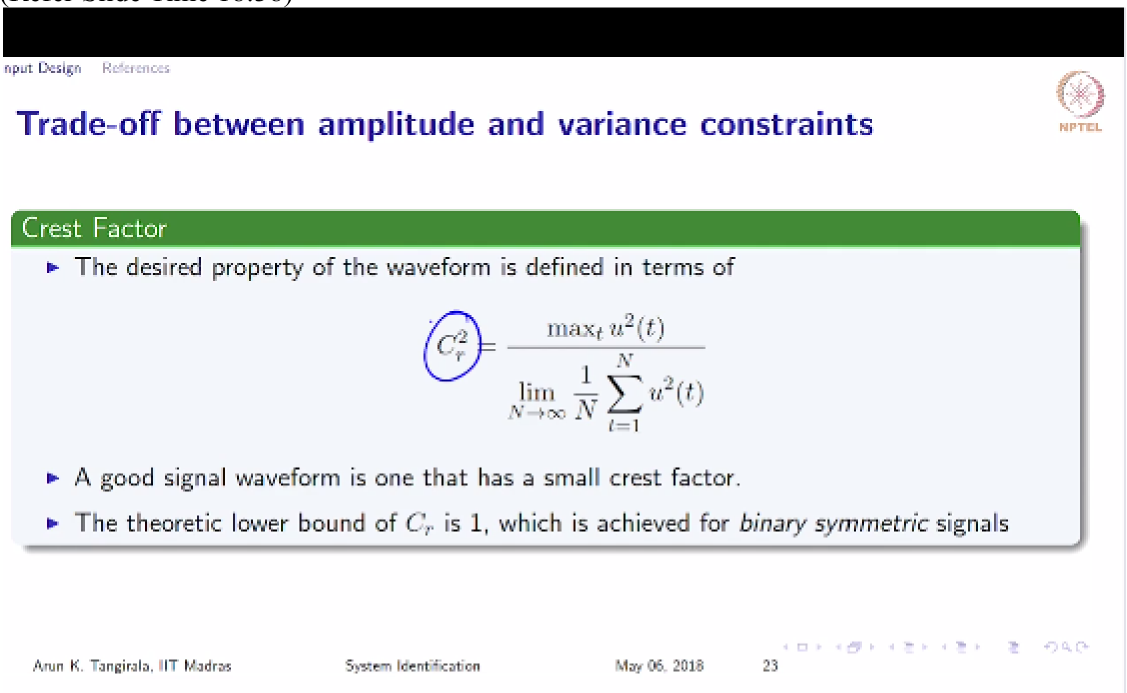
So straight away it tells you that you have to set up any equations to estimate the n unknown bs and the terms in your n by n matrix will be essentially the variances, the covariance such as this, right? In fact, this should be. It's a mistake. I'll correct. So you will have variance along the diagonals, auot-covariances along the half diagonals, and there you go. The matrix has to be a full rank. So you can even discard or disregard for now the entire theory that you have learnt of persistent excitation. But it's hard to do that because we are saying it is persistently exciting of order n. But this is an alternative way of looking at persistent excitation. You don't have to turn to frequency domain. You can take this as the definition of persistent excitation of order n. Now you can extend it to general persistent excitation, which means that my input should have sufficient excitation that I should be able to estimate FIR models of any order. That is another way of looking at the general concept of PE, persistent excitation.

(Refer Slide Time 16:30)



Okay. So now that we have understood what is a persistent excitation, we can look at other aspects of the input design. What we have learnt is one very important aspect of input design. The other aspects to keep in mind is that the asymptomatic properties of the estimate which estimate the parameter estimates that depend only on the input spectrum, not on actual waveform. This is true for linear systems, which means whether I use sinusoidal mix of sine PRBS and so on, it really does not matter to the bias or the variance of the parameter estimate. It only says, tell me how much weight even -- how much power is there at a certain frequency. It doesn't care whether you've used a sine wave form or a PRBS wave form. But on the other hand, we want the input to be of limited amplitude. Why? Why can't we have input of high amplitudes? Because we do not want to push the process into nonlinear regimes. We don't want this. High amplitudes tend to push the process into nonlinear regimes. The basic premise in lot of applications where we build linear models is that the process is approximately linear in a small neighbourhood. So if you excite the system with large amplitudes, you may end up pushing the process into non linear regimes. So we don't want that. But we also, although I've not listed here, a very important thing is that the SNR plays a very, very important role, signal to noise ratio. Which means I want to maintain good power in the input, I want to maintain good power at all frequencies persistent excitation. When I say good power at all frequencies relative to noise. Suppose the noise levels are high, then I have to excite the system even more but then I run into the risk of pushing the process into nonlinear regimes. So there are some conflicting requirements here. And these conflicting requirements are captured somewhat in a measure called crust factor.

(Refer Slide Time 18:38)



So the crest factor is a ratio of the maximum amplitude that you have, of course, it is squared amplitude that you're looking at. And the total power. And we want an input that has the smallest crest factor. So why is that? We want to pack the maximum power for a given amplitude, for a given maximum amplitude, that is, different input waveforms can correspond to the same power. Power means variability, whatever you can think of it. So for a given power, there may be different input waveforms. But I want to pick the waveform that has the lowest maximum amplitude, so that the process is not pushed into nonlinear regimes, right? Typically, if I want high power, I want high amplitude. But what we -- what I've just said is for a given power, there are many waveforms. So if I choose unit power, I can generate a unit Gaussian signal, random Gaussian signal which has unit, let us say, unit variance. I can choose a random Gaussian signal that has unit variance. I can choose sinusoidal signal that has unit variance. I can choose a binary, random binary signal that has unit variance. You will notice that each of them has a different maximum value which is there in the numerator, right? In fact, what is the maximum value that you expect to see roughly for a Gaussian white noise signal of unit variance? Typically, standard Gaussian distribution, you will see plus or minus 3, right? Roughly that is the maximum value that you will see. But the overall power is only, variance is only unit e. It turns out that you could achieve, you could pack the same variability or the same power with a lower peak amplitude when -- if you use random binary signals or PRBS. And that is why the PRBS signals are preferred to the white noise signals because they have a low crest factor. In fact, all binary signals, binary symmetric signals have the crest factor is 1 which is the lowest.