

MITOCW | 13. Frequency response of LTI systems

The following content is provided under a Creative Commons license. Your support will help MIT OpenCourseWare continue to offer high quality educational resources for free. To make a donation or view additional materials from hundreds of MIT courses, visit MIT OpenCourseWare at ocw.mit.edu.

PROFESSOR: So today I'm going to continue with frequency response and filtering, but also begin the story of spectral content of signals. So our starting point is still something you've seen before, namely the statement that for an LTI system, a sinusoid into the system gives you a sinusoid out at the same frequency, but maybe shifted in phase and scaled in amplitude.

So a bit of terminology here just for general interest, we refer to the exponential as an eigenfunction of the LTI system, because the only effect the LTI system has on it is a scaling. So an input to some kind of a mapping, which comes out the same except for a scaling is referred to as an eigenfunction, or an eigenvector if you're talking about matrices. So we say that the exponential-- the complex exponential here is an eigenfunction of the LTI system.

Because when it comes through, it's just the same exponential, but scaled by some number. And that number is what we refer to as a frequency response, right? And we've seen that there's a simple expression for it. And let me put that expression up, because we're going to use it repeatedly.

The m here is irrelevant. It can be any dummy index, because we're summing over the m . You can call it anything you want. And I just should mention that there's other notation for this object. There are people who refer to it as-- well, it's often referred to as h of $e^{j\omega}$, because actually, the way ω enters is always in the term e to the $j\omega$. So this is-- if you want to think of it that way, this is e to the $j\omega$ to the power minus m , right? Well, let me just write it as 1 over 1 over e to the $j\omega$ m .

OK, so it's some function of e to the $j\omega$. And people will often write it this way. And one of the advantages of this is the notation right away tells you that this object is periodic with period 2π . Because if you were to increase ω by an integer multiple of 2π in the numerator here, you'd get the same argument again. And therefore, h must be the same again.

So this notation has the value that it keeps the periodicity front and center. It also makes sense when you're developing various other transforms. There's something called a z transform, which we won't deal with in this class. But it's used a lot when dealing with discrete-time systems. And the way that you get from the z transform to this object is by making the substitution z equals $e^{j\omega}$. So people will use this notation.

So the z transform uses z exactly where we use e to the $j\omega$. But for our purposes, this is a much simpler notation. It's just that we need you to remember when you see this that we're talking about something that's got period 2π . And if you look at the definition, that becomes clear. If you increase ω here by any integer multiple of 2π , you're going to get the same thing back again.

There's another bit of notational confusion that can arise, which is that people will sometimes write little ω instead of big ω . So that's also used. So this is other notation, and it's notation that we will try not to use, but you might see vestiges of this when you look through old problems. Because in some terms, we may have used this notation, and some terms, we may have used a little ω instead of a big ω . But for our purposes, we'll stick to this.

OK, so when we say big omega, we're thinking of it as an angle around the unit circle. So if you've got the complex number here at an angle big omega, this complex number is $e^{j\omega}$, right? So we're thinking of big omega as an angle, something measured in radians, and it's different from little omega.

You can write the expression for the frequency response in various ways. So here, I've just used Euler's identity to split that into a cosine and a sine, and that's straightforward enough. The sums are over infinite intervals. And we talked last time about how stability of the system-- bounded input, bounded output stability of the system will guarantee that those summations are well defined.

OK, now there's another name for this formula. Basically, we've called it the frequency response. But when you compute an h of omega using this formula, another way to say what you're doing is to say that you're taking the discrete-time Fourier transform of the sequence h dot, OK? So it's the discrete-time Fourier transform. Again, that's just terminology for now. We'll come to expand our view of it later.

But we've called it the frequency response so far, because it describes how sinusoids or exponentials here get to the output, but it's also referred to as the discrete-time Fourier transform of the unit sample response. So you've got some time signal-- happens to be a unit sample response. You compute an object through this formula to get an h of omega. That's the DTFT, OK?

Another thing we've already seen is that knowing that you have an LTI system, and that a cosine is a superposition of complex exponentials, you can use the result that we had so far to just describe what happens to a cosine when it goes through the system. So it's no longer a complex exponential. It's a real signal of the kind that we're more likely to work with. And we've seen that the only thing that happens is the cosine that went in gets scaled in amplitude by an extra factor, which is the magnitude of the frequency response. And whatever phase it had, you get an extra phase, which is the angle of the frequency response.

So actually, if you had an LTI system, this is a good way to measure the frequency response in the lab. What you do is you take your system there, excite it with a sinusoid. In continuous-time, we know we can do that with an oscillator. In discrete-time, you generate a sequence like this. And then look to see what comes out of the system and express it in this form, and you'll label the scale factor there as the magnitude of the frequency response and the extra phase angle as the phase angle of the frequency response. So it makes for a very systematic way to probe a system and get at the frequency response.

Again, of a point that I've made before, which is that when you do this probing, you only need to vary big omega over the range minus pi to pi. So when we write a frequency response, because h of omega is periodic with period 2π , we only need to probe h of omega-- either the magnitude, that would be one plot, and the angle would be another plot. Both of these would be plotted from minus pi to pi.

Because outside of that range-- well, you can see it already with the cosine. If I added an integer multiple of 2π to omega 0, I'm going to get an integer multiple of 2π added into the argument of a cosine. And I'm getting the same cosine back again. And the reason that's the case is because the n that's multiplying it here is an integer. So in continuous-time, it doesn't work quite the same way.

If I had a little $\omega_0 t$ and I added multiple of 2π , I wouldn't get the same argument back again. So what's different here is that if I increase ω_0 by an integer multiple of 2π , because I've got an integer, and outside that, again, I end up adding an integer multiple of 2π , and I'm back at the same cosine. So frequency response for a discrete-time system is always in the interval $-\pi$ to π . It repeats periodically outside of that if you chose to look at some other ω .

And I've said that already. You actually heard the term frequency response in all sorts of settings, I'm sure. One setting in which it's used a lot is in describing, for instance, the performance characteristics of a loudspeaker. So people will tell you how good their loudspeaker is by showing you the frequency response of the speaker. And what they're doing is they're applying a sinusoidal voltage to the input and looking at the sound pressure that comes out.

SPL here is sound pressure level. This is measured in dB, so it's actually a measurement of the ratio of the pressure that you hear under certain standardized conditions to a pressure which is taken as the lowest audible pressure on the ear. So there's a particular ratio there.

So what they'll do is they'll feed the loudspeaker with 1 watt at 1,000 Hertz, so just a steady tone. And then, a meter away from the speaker in an anechoic chamber, they'll look to see what sound pressure they pick up on a specialized sensor-- a detector, a microphone basically-- and that number in dB is what they'll represent. And so typical speakers are-- have values in that kind of range.

Now, if you probe it at different frequencies applying the same input voltage and looking at pressure, you'll get varying pressure depending on the frequency that you probe at. So this is the frequency response of the speaker, and if you go too low in frequency, then you don't get much of a response. If you go too high in frequency, you don't get much of a response.

Now, of course, when you use the speaker, you're not going to probe it with sines and cosines. You're actually going to put more complicated sounds in there. So what you're really interested in is how does the speaker behave to signals that are combinations of cosines? And again, we're using our model of the speaker as an LTI system. All bets are off if you drive your speaker so hard that you get distortion and exercise all the nonlinearities there or burn it out.

But if you're in a normal range, the speaker is acting linearly, you can talk about its frequency response. And what you're really interested in is how does the speaker respond to linear combinations of cosines? And all of these various signals can be thought of as-- at least over reasonable time intervals-- as combinations of cosines appropriately chosen. So if you hit a particular key on the piano, you get a dominant note, but you'll get harmonics of that. And that's what's going into your speaker.

So knowing how an LTI system responds to cosines then puts you in a position to say how it responds to combinations of cosines, or signals that are combinations of cosines. So the other part of the story that we're going to get to-- and maybe even by the end of this lecture-- is we need a way to take a general signal and represent it as a combination of cosines. And that's what we refer to as the spectral content of the signal.

So when we talk of exposing the spectral content of a signal, as over here, what we're saying is we're going to show you what combination of cosines it takes to make up that signal. And once you figure that out, and you have the frequency response of your LTI system, you can say how your system responds to that. OK, so this theme runs through every stage of what happens, actually, in communication.

Now, the example I've given you here is one that you would typically probe with a continuous-time oscillator in the lab. And so there's some connections that you might want to make between probing with a continuous-time signal and probing with a discrete-time sequence that comes from sampling that signal. But I'm going to leave you to look at that later or leave your recitation instructors to pull that back, or leave you to draw this up if you have a homework problem that needs you to think about how continuous-time maps to discrete-time.

But the basic point is the actual, physical speaker you might probe with a cosine in continuous-time, if you're generating that signal from a computer, what you'd actually be sending to your amplifier is a sequence of numbers. And the frequency of the numbers that you would send, this frequency is related to the frequency of the continuous-time cosine that you want in a very particular way. So I'll leave you to chew on that. But I don't want to spend time on that now.

OK, so let's spend a little time talking about the properties of frequency response now that we know why we would use it. And this I've already said. The value of the frequency response at-- some of this, by the way, you may have seen in recitation. But it doesn't hurt to repeat. The frequency response at frequency 0-- well, we've said if you've got $e^{j\omega_0 n}$ and some frequency ω_0 going into a system $h(\omega)$, a system with frequency response $h(\omega)$, an LTI system with frequency response $h(\omega)$ -- all right, I'm leaving out lots of words. But frequency response doesn't make sense unless you have an LTI system.

OK, so for what kind of input signal would you be looking at $\omega = 0$? DC, right-- a constant signal. It's what the electrical engineers call DC, which used to stand for direct current but has now come to mean constant. When we say a DC input, we just mean a constant input.

So if I pick ω_0 to be 0, then $e^{j\omega_0 n}$, well, that's just one for all time. And so I'm feeding the system with a constant. That's the slowest possible input that you can find. It's a 0 frequency input. And the amount that it's scaled by is the number that you're going to plot here. So whatever value you get is going to end up being plotted there at $\omega = 0$.

And let's see, do we believe this other statement-- $h(0)$? It's just a substitution in here. If I put $\omega = 0$, it's a summation of all the $h(m)$'s. But there's another way to think of it also. If you want to think of it in the time domain-- let's see. I have an LTI system. It's got some unit sample response. And I'm feeding it with an input that's constant for all time. It's actually constant at the value 1 for all time.

If you're thinking in terms of convolution, the flip slide and dot product picture, what is the output at any time here? You're going to draw out your unit sample response. You're going to draw out your input, which is 1 for all time, take one of them and flip it over, slide it the appropriate amount over the other, and then take the dot product. Well, for every shift of this flipped and shifted input, you're going to pick up all of the unit sample response. So every time, you're going to get summation $h(m)$ outside.

So if you fed an input that was DC at the value 1, this is what the output will be at all times. You can see that from the convolution picture. So what's the frequency response at frequency 0? What's the ratio of the output to the input-- the output amplitude to the input? This is for all time. So the input amplitude was 1 at each time. The output amplitude was that. And so that's the DC gain-- the DC gain of the system, or the frequency response at 0. Let's say, so h_0 is what's referred to as the DC gain.

What about high frequency? So what's the highest frequency variation that you can have with a discrete time sequence? I've got a sequence here at the input. We've seen what the slowest variation possible is. It's something that's constant. If you're talking about a discrete-time signal that can only take values at integer times, what's the highest frequency variation that you can get?

Just something that alternates in sine, right? So you're going to have-- OK, so is this of the form $e^{j\omega n}$ for some ω ? Is that a signal of exponential form? Yes?

AUDIENCE: [INAUDIBLE]

PROFESSOR: Yeah, if you take ω not equal to π , this is just $e^{j\pi n}$. In fact, you can take plus or minus π . So when you probe the system with an input of this type, which is the highest frequency input that you can probe with, what you're really probing is what's the frequency response at this point? You get the same value at minus π or π .

So these are the two extremes. And then the frequency response, the rest of it lies in between for other sorts of inputs. Now, do you believe this other identity that I have up there? Well, you can go back to the definition, set big ω equal to π or minus π , and you'll get an alternating sequence of 1's and minus 1's here. And so that verifies that identity.

Or you can think in terms of convolution. If I convolve a sequence like this with a system with this unit sample response, what comes out at every time is an alternating sum of the h_m 's, except the sine flips from one time to the next. And so, again, you can verify in the time domain that that's actually the high frequency gain of the system, OK?

Now, there's a bunch of other symmetry properties of the frequency response that I think in-- at least in some of the recitations you've done. And the easiest way to see these symmetry properties is to actually go back to the rewriting I did of the frequency response in terms of sines and cosines. This first term here I'm calling C of ω . The second term here, the summation, I'm calling S of ω .

So where would a statement like this come from? Let's see. For real h of n , that's the only kind of h of n we're going to worry about in general. We're going to talk about systems with real unit sample responses. If h is real, why would it be true that the real part of the frequency response is an even function of frequency?

Well, the real part of the frequency response is this term, because the other term is the imaginary part. So the real part of the frequency response is this term. And if I change big ω to minus ω , the cosine doesn't change. It's the same. And therefore, the real part is even, OK? So the real part of the frequency response is an even function of ω . The imaginary part, which is the minus S of ω , well if I change ω to minus ω , I flip the sine. So that's an odd function of ω , and so on.

So you can go through these properties. Whenever you're stuck trying to figure out a property, this is the expression to go back to. So rewrite the basic definition in this form, and you'll understand a lot of this. And again, you'll get practice in recitation if you haven't done that already.

Another important property of-- that you encounter when you go from the time domain to the frequency domain-- so remember, in the time domain, we said that if you have an input here, you convolve that input with h_1 to get the output of the intermediate point? OK, so if I call the output of the intermediate point-- I should have done it there. But here's h_1 . If I call this w , this is x . And then I go into a second system, h_2 . And here's y .

OK, well w is equal to h_1 convolved with x . And y equals h_2 convolved with w . So that's this. But I can put the parentheses any way I like for convolution, right? We've already established that property. So the net effect of the cascade of systems is the effect you'd get by having a single system LTI with this unit sample response.

Now, if I think the frequency domain-- if I put $e^{j\omega n}$ here, then what comes out at the intermediate point? At the intermediate point, I get $h_1 e^{j\omega n}$. All right, so that's w_n . But this is, again, an input of exponential form. So what comes out of the second system when I put this input into it? So what's w of n -- sorry, what's y of n going to be? Yeah?

AUDIENCE: [INAUDIBLE]

PROFESSOR: Yeah, so it's basically the second system's frequency response scaling the exponential that went into the second system, which is this. So the net effect when I put $e^{j\omega n}$ in at the first spot is at the output, I get the same $e^{j\omega n}$, but scaled by the product of the two frequency responses. So the nice thing here is that when I'm describing a cascade of two systems, if I describe the net effect in the time domain, I've got to do a convolution of these two units' sample responses. If I think of it in the frequency domain, I just have to take the product of the individual frequency responses.

So the key observation here is that convolution in the time domain maps to multiplication in the frequency domain. So if I wanted the DTFT of this-- if I wanted the discrete-time Fourier transform of this result of a convolution, I can find it by just multiplying the individual DTFTs, all right? So convolution in time maps to multiplication in frequency.

And this actually makes design much more easy, because we're often cascading systems in this form. And if you think in terms of frequency, you can track a frequency component through a cascade of such systems just focusing on the frequency response of each system as you go. So here's an example. Suppose we have a channel. Let's say that it's a channel with an echo, so when I put-- let me actually draw it out here.

So I've got a channel here which I'm modeling as LTI. And if I put in a unit sample function here, so this has the value 1 at time 1, suppose the channel is one that has some echoing in it. So what I actually get out for this input is the same $\delta[n]$ plus $0.8\delta[n-1]$. So there is a later arrival scaled by something which corresponds to the echo. So this must be the unit sample response of the channel.

What's the frequency response of the channel? So if I call this $h_1[n]$, what is $H_1(\omega)$ -- big ω -- $H_1(\omega)$ of big ω ? I don't have it up there, do I? No. Anyone? Just from the definition. Is the problem here that you don't quite see what $h_1[0]$ is, $h_1[1]$, $h_1[2]$, and so on? If I asked you to plot this out, how would you plot it? Yeah?

AUDIENCE: [INAUDIBLE]

PROFESSOR: Is it 1.8? Where? Where would you put the 1-- just over there? Oh, you're talking about the frequency response. Let's get the unit sample response first. Let's sketch this. What's your sketch of that?

AUDIENCE: At 0B1

PROFESSOR: At 0B1?

AUDIENCE: [INAUDIBLE]

PROFESSOR: OK, on 0 everywhere else-- that's the unit sample response. OK, so what's the frequency response? Well, we just plug it into the definition. All the h's except the ones that argue in 0 on 1 are equal to 0. So this is going to be 1 plus 0.8 to the minus j omega. Is that what you said? It was not quite what you said, right? What you said was the number I'd get at omega equals 0-- the DC gain of the system. But the frequency response is that.

Let's just work backwards here. So the frequency response is that. Or if I wanted to write it-- we're going from that board to here-- h_1 of omega, I can write it as a real plus imaginary part. So it would be 1 plus 0.8 cosine omega. This would be the real part. Then I have a minus j-- sorry, 0.8 sine omega. OK, so that's the frequency response-- some complex number with a real part and an imaginary part.

OK, and if I asked you to give it to me in magnitude and angle form, you could do that. It's just rearranging things. So you'd-- the magnitude would be the square root of the sum of squares of these two pieces. And the angle would be the arctan of the ratio. So I assume that you know how to do all of that.

And what you find-- actually, you can see it in these expressions already. Just as-- well, I didn't quite claim this earlier. But the magnitude of the frequency response will always be a real function of frequency-- sorry, an even function of frequency. And the phase will always be an odd function of frequency. So if you're drawing the results of a computation like this and you find that you don't have an even function for the magnitude, then you know you've done something wrong.

So I'm not-- I'm going to sketch something here which I'm not pretending is the magnitude of that. I just want you to get the idea of what I mean by even. It's going to be something that's symmetric in omega. This is the magnitude. And then, if I did the phase, the phase is always going to be something that's an odd function of frequency.

So if it's an odd function of frequency, what's the value at 0 of the phase? It's got to go through 0, right? And so I might get-- well, what would it actually be? It would be some shape. I'm not pretending I have the right shape here. But it's going to have an odd symmetry. I'll leave you to figure out what it actually looks like. So that's the frequency response of this echo channel.

So here's what I want you to do now. At your receiver, build for me a filter that's going to undo the distortion that the echo has produced. So what I'd like is, I'd like an output, after you've done your filtering, to be exactly equal to the input. So my question is, what should-- and my claim is you can do that with an LTI filter. How would you describe that LTI filter? What should that LTI filter be? Yeah?

AUDIENCE: [INAUDIBLE]

PROFESSOR: Right, OK, so if you wanted the output to be exactly equal to the input, no matter what input was, you want a frequency response of 1 overall. And the overall frequency response we know is the product of the two individual ones. And so we want $h_2 \omega$ times $h_1 \omega$ to be equal to 1. And therefore, h_2 should be 1 over h_1 .

So you can see here how things get a lot easier when you think in the frequency domain. If I had to do this in the time domain, I would have had to say h_2 convolved with h_1 has got to give me the unit sample function. And I'll give you h_1 , now you've got to figure h_2 . Well, you've got to go and work the convolution picture backwards, which is doable for simple cases. But this is much simpler. So this shows that h_2 should be 1 over h_1 .

Seems like a reasonable way to go. And you can actually work the whole thing through. But there's a problem with this. And we've seen this in other settings as well, which is something that works fine in the noise-free case doesn't work so well when you've got noise in your system.

So look at what this receiver filter is doing. The receiver filter-- let's see, what is its magnitude? How does the magnitude of the receiver filter relate to the magnitude of the channel filter-- of the channel frequency response? So this magnitude is a magnitude of 1 over h_1 . Is that the same as 1 over magnitude of h_1 ? Is that how complex numbers work? OK, right?

So look what happens. Where the channel has a very low frequency response-- in other words, where the channel output is very low for a sinusoidal input at that frequency, the receiver filter is going to have a very high magnitude. So the receiver filter is trying to boost up whatever signal it sees in a frequency range where the channel actually has very little output.

So what happens if I come and have a bit of noise here where I'm receiving the signal? Well, it's going to be very badly exaggerated by the inverse filter. So a little bit of noise here will get accentuated at frequencies where the frequency response of the receiver filter is large. But that's precisely where the channel had a very low frequency response. And it's precisely where the output-- the channel-- has nothing interesting for me. So my receiver filter ends up accentuating the noise.

OK, so yet again, we see that these sorts of inversion operations may look nice on paper. But if you don't take account of what noise does, then you can run into trouble. And the picture is very transparent when you think in the frequency domain.

OK, some more practice with filters and cascade-- I think I'm going to leave you to work through this in recitation, perhaps. So I'll leave it on the slides. But let's go to design of filters. So now, we've seen one example of trying to design a filter-- the receiver filter-- to undo the distortion of the channel. Here's another-- actually, I want that. Here is another design problem that you run into all the time, which is that you see a signal that's got a whole bunch of frequencies mixed up in it, and you want to exclude some of them.

So maybe you're looking for an audio signal. You know that the combinations of sinusoids that make up an audio signal are unlikely to go above-- whatever you want. Pick your number-- 10 kilohertz, 20 kilohertz. And so you want to exclude frequencies outside of that range. So you're very often in the position of trying to build what's called an ideal low pass filter.

So here's an ideal low pass filter. I'd like you to build for me a filter that passes all frequencies in some range without distortion, and that completely kills everything outside. So let me call this the cutoff frequency. So that's the h of ω I want. And now my question is, how are you going to build this filter? I want you to give me the unit sample response that goes with it. And you see a hint over here. But can you tell me how you might go about that?

Not so obvious, right? Because we've specified the filter characteristic in the frequency domain, and now we want to find the h 's that go with it. So what we're really looking for is a formula that will give us the time domain signal in terms of the frequency domain. So we want to invert this somehow. So what we're looking for is really what's called the inverse DTFT.

And actually, if you've done Fourier series, you've seen this trick before. Because really, we're not far from Fourier series here. It's just that the domains are a little different, so maybe you don't recognize it. Here, we've got a periodic something expressed as a combination of sines and cosines, or as a combination of exponentials. And now, we want to invert that, OK?

If you thought of these as Fourier coefficients for some periodic signal, and then went and looked up whatever book you use for Fourier series, you'd get the formula. Because we're just trying to extract the Fourier coefficients for this periodic signal. But you can actually do it from scratch.

So if you think of multiplying both sides of this by, let's say, e to the $j\omega n$ -- OK, so I'm going to multiply both sides. So I've got e to the minus $j\omega m$ minus n now, right? And I'm going to then integrate both sides over an interval of length 2π -- any contiguous interval of length 2π . It actually does matter because of the periodicity. So I'll take any interval of length 2π and I integrate both sides.

And I'll assume that I can hop this integral in there. I'll assume my signal is well-behaved enough for that. So here's what I end up getting. On this right hand side, I get summation integral h_m . Oh, I should put a $d\omega$ there. Sorry. I've gotten casual with my integration.

So on this side, I have this integral. On this side, I have that integral. And if you work through this, out of all this infinity of terms, there's only one term that survives. Because any term in which m is different from n will have this exponential still sitting here. This exponential is like a cosine plus a j sine, or a cosine minus a j sine. You're integrating it over an interval of 2π .

So any term here that has the exponential, or has the sine or cosine in it, will disappear under the integration. The only term that survives is the one where m equals n . And so what you discover is that this is $2\pi h_n$ when you're all done. I'm not going through the details here. So here is the formula we wanted for the inverse DTFT. Here's the inverse DTFT, OK? I've forgotten my colored chalk today, but that'll do.

So if I gave you a filter characteristic like this and asked you to find the unit sample response of the filter that went with it, you would just have to plug in the frequency response characteristic that I gave you and solve for the h 's. I think I have a bunch of this on the slides. This is what we just went through. So let's do this now for the ideal low pass filter.

What is it that we do? I've got the formula that I just derived for you there. h is equal to 1 in the pass band of the filter, and it's 0 outside of that. So I set h equal to 1 in the pass band of the filter, which is from minus ω_c to plus ω_c , and the rest of it doesn't contribute anything. And then, I just work out this integral.

And I've actually got to do it in two pieces. For n not equal to 0, this is what I get. For n equals 0, this is what I get. If n was continuous, actually, you'd say that this is the same expression as here, because you just use L'Hopital's rule and you'll get from here to here. But since n is an integer, we've got to be a little careful how we write it, OK? So you can't really say you're going to use L'Hopital's rule to see what this is in the limit of n going to 0, because n takes integer values. But if you work it out from scratch for n equals 0, you'll see that you get a formula that's consistent with using L'Hopital's rule.

OK, so this is a function that we'll see again and again when we do filtering of this type, and it's referred to as a sinc function. So it's not S-I-N, but S-I-N-C. And if you plot it out, this is what it is. So it's got the oscillation that comes from the sine, but it's got a reduction in amplitude that comes from the $1/n$. So it's a signal that falls off as $1/n$ with this kind of a characteristic.

Do you think it's a bounded input, bounded output stable system? What's your hunch? Remember what it takes for a system to be stable? The unit sample response has to be absolutely summable. So if you take the absolute values here, and sum from minus infinity to infinity, you want to get something finite to call this stable. Well, since this only falls off as $1/n$, it turns out to not be stable. So it's actually an extreme idealization that is not bounded-- input bounded, output stable, but it's close.

Just to go back when I showed you this filter characteristic here, to give the cheap version of a low pass filter, what we actually did was take the sinc function and truncated to a finite interval. And so what happens when you truncate it to a finite interval is that instead of the sharp box-like shape for the frequency response, you get a closer approximation to it-- not exactly the ideal low pass filter, but maybe good enough.

The other thing that you might notice if you're looking carefully is that I had a sinc that was centered around 0 and even. And now, I seem to have a causal version of the filter. And I think I'll leave you in recitation to figure out how you can go from the centered, non-causal filter to a causal filter, and what that does to phase and to frequency response magnitude. So basically, I'll leave you to go through the details here. But the key idea here is the inverse DTFT.

So now, I want to just take a slightly different perspective on this formula that we derived. We said we've got a frequency response, which we're calling the DTFT of the signal h of n -- the unit sample response. We've got an inverse formula that allows us to get the time signal from the frequency response. But here's yet another way of looking at what this formula is telling us.

This formula is saying, I can think of h of n as being made up of a whole bunch of complex exponentials. So you see that this is what we were looking for. We were looking for a way to take a signal and figure out its spectral content. We want to know what complex exponentials, or what sinusoids does it take to make that signal?

Well, we have a hint of that in this expression, because this is saying, take the time domain signal. I can think of it as being a combination. Now, this is not a finite combination, it's a continuum. But it is a combination of exponentials of the type that we know to work with. So this is actually giving us a spectral decomposition of the unit sample response, where the amount of e to the j ωn that it takes to make up the signal is told to me by h of ω .

So the h of ω s are sort of the weights that we use to combine these exponentials to get the signal. So the idea for a spectral decomposition, or for describing the spectral nature of a signal is actually sitting there. All we have to do is say, we'll use the same formulas, but let's no longer restrict it to the unit sample response of a system and the frequency response of that system.

Let's use it for any signal-- so the same formulas, but now for any signal x_n . Give me any signal x_n , I'll compute for you this object, which is the DTFT of that signal, just the same way I did for a frequency response.

So I'll compute the x of ω for you. What's the significance of x of ω ? Well, it tells me in what combination I have to wait the e to the j ωn 's to construct for you the signal. So the x of ω , the DTFT tells me what the spectral content of the signal is. If I plot that as a function of frequency, it tells me how to assemble the signal out of sums of sines and cosines.

So let's see here. More specifically, what I would say is that the DTFT at ω_0 -- ω_0 times $d\omega$ is the spectral content of the signal in that particular interval. And if I add up all those components over all frequencies in this 2π range, I'll get the original signal that I'm interested in. So what we'll do next time is work with this idea to see how it lets us think about signals through systems, and how it enables us to do filtering in a systematic way. All right, let's leave it at that for now.