

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.

DENNIS So hello and welcome. As I mentioned last time, we're essentially done with the course. We've
FREEMAN: done all the theoretical underpinnings.

What remains is to talk about two important applications of Fourier, in fact some applications that are very difficult to do if we didn't have Fourier analysis and in fact quite simple to think about once we have Fourier analysis. So today I'm going to talk about sampling. We'll spend this lecture and the next lecture on sampling. And then the following two lectures will be on modulation and we're done. So we're almost done.

So we've talk about sampling lots in the past. In fact, it was on the very first homework. That's in fact I think one of the strong points of this course is that we regard continuous time signals and discrete time signals on equal footing. And part of the goal is to be very comfortable to convert back and forth, because both representations are so important.

We see CT coming up in fundamental ways, because a lot of the things that we're interested in are systems based on physics, and that's just the way it is. Physics works in continuous time by and large. However, because of digital electronics, we like to process things with digital electronics, because it's so inexpensive.

So for that reason, we want to go back and forth. And what we'll see today is that when we think about system levels, when we think about signal level conversion, Fourier transform is the key. So keep in mind we've already thought about how you would convert a CT system into a DT representation. We did that back in about Homework 3 or so.

So what's special today is thinking about-- rather than thinking about systems, thinking about signals. And as you can imagine, there's enormous reasons why you would want to think about signals from a digital point of view. Virtually all the things that you play with all the time are digital.

So you think about audio signals, they're now stored digitally. Thing about pictures, digital. Video, digital. Everything on the web, because there's no other way the web can work. If it's on the web, it's digital.

So there's just an enormous reason why we would like to understand how to take a continuous

time signal and turn it into a discrete representation. This is just motivation. We tend to think about common signals that we deal with everyday as though they were in continuous time, continuous space, same thing.

We think about things like pictures as though they were continuous. They aren't. Anybody who has a digital camera knows that if you zoom in enough, you see individual pixels. They are not continuous representations. They're discrete representations.

Even some kinds of pictures that are ancient-- well, ancient by your standards at least. Even well-known kinds of pictures like newsprint, in an underlying sense, they are discrete representations. So what's showed here is a picture of a rose and a halftone image of the type that would be printed in a newspaper.

And if you zoom in, you can see this-- so if you zoom into this square so you can see this better, this, which is not really continuous, I'm showing it on a digital projector, it's actually got pixels too. But the pixels are small enough for the time being I'm going to ignore that. So consider this continuous even though it isn't.

And you can see the discrete nature of this one much more clearly. In fact, the halftone pictures that you see in a newspaper are not only discrete in space, but they're discrete in amplitude, because they're printed with ink. Ink comes in two flavors, ink or no ink. So they are a binary representation in intensity as well. And we'll talk about that a little bit more the next time.

So in order to have a complete digital representation, you need to think about not only sampling in the time or here the space dimension, but also sampling in the voltage or the amplitude dimension.

Even the highest resolution picture you have ever seen is digital. So this refers to the completely ancient technology of how do you make a digital print? So a very high-quality picture is made from an emulsion of some sort of a chemical, originally silver bromide. The idea was that you had very small crystals of silver bromide that could be reduced by a photon to turn them into silver metal.

And the idea was that exposure to light would therefore convert silver bromide to silver. And then developing meant washing away the silver bromide salt that remained that was not converted, leaving behind the silver that had. And that was the basis for the chemical reaction

that gave rise to pictures. The point being that even there, these crystals are on the order of a micron in size, and they're either on or off. So even there it was a sampled version.

And if it weren't enough, everything you've ever seen is sampled, because that's the way your eye works. Your eye has individual cells that either respond to light or don't. There's about 100 million rods, about 6 million cones. So every image you have ever seen is sampled.

So one question is-- and it's such a good sampling that you don't even notice. But maybe that's because you're, well, unaware, to be polite.

So think about, how well sampled is it? So I know that this picture is sampled, because I can come up and I can see the individual pixels. I can see a little grid of pixels. There's 1,024 by 768.

I want you to think about how well your eye is sampling that by thinking about whether or not you should be able to see the pixels from where you're sitting based on the sampling that's in your retina. So look at your neighbor, say hi. Figure out whether you have enough rods and cones to see individual pixels or not.

OK. Does anybody-- so who can tell me a way to think about this? Or who can tell me the answer? Do you have enough rods and cones to sample the pixels on the screen? Yes?

AUDIENCE: No.

DENNIS No.

FREEMAN:

AUDIENCE: I can't see them.

DENNIS You can't-- you cannot.

FREEMAN:

AUDIENCE: I don't see the pixels.

DENNIS OK. Well, that could be because you don't have enough rods and cones.

FREEMAN:

[LAUGHTER]

AUDIENCE: I have slight astigmatism.

DENNIS Ah. Astigmatism. Is that a rod and cone problem?

FREEMAN:

AUDIENCE: That's a lens problem.

DENNIS That's a lens problem. So maybe you have enough rods and cones and not enough lens.

FREEMAN:

There's actually another reason you might not be able to do it, besides rods, cones, and lenses.

AUDIENCE: Your brain.

DENNIS Brain. There's even another one. OK. We're up to-- there's another reason why you might not

FREEMAN: be able to do it. Rods, cones, lenses, brains.

AUDIENCE: Photons.

AUDIENCE: Photons?

DENNIS Photons. That's an interesting thought. I think you could probably pull that off, yeah. The

FREEMAN: number of photons-- if the lighting were low enough-- your eyes are very sensitive. You can see-- you can see-- you can report a difference with one photon, one. It's pretty little. It's kind of the limit, ? right? Yeah?

AUDIENCE: It's so far away that I can't see the pixels.

DENNIS It's so far away that you can't see the pixels. But why? Is it because you don't have enough

FREEMAN: rods and cones, because your lenses are screwed up?

AUDIENCE: [INAUDIBLE].

DENNIS So you might be using your rods and cones for different things. Your cones are focused in an

FREEMAN: area called the fovea, right? So one way you could improve that would be to look at it.

Can somebody think of something besides rods, cones, lenses, and brains? Can somebody think of convolution lecture with some sort of application that we did in convolution? No, of

course not. That was more than 10 lectures ago.

In convolution, we looked at the Hubble Space Telescope and we looked at a microscope.
Yes?

AUDIENCE: There's going to be tons of particles in the air, so--

DENNIS
FREEMAN: Particles in the air. That was something that happened in Hubble. So smoke-filled rooms, that's bad. From the Hubble experiment, from the Hubble lecture we talked about how there was a point spread function associated with diffraction. And there's also a diffraction limit because of the size of your pupil. Because you're looking through a narrow aperture, that limits the resolution as well.

But let's get back to this, rods and cones. You have enough rods and cones. How you do that? How do you think about whether you have enough rods and cones?

OK. Step 1, look at the previous slide. What was the important thing on the previous slide?

AUDIENCE: [INAUDIBLE]?

DENNIS
FREEMAN: Three microns per rod and cone, right? So rods and cones are separated by three microns. So what do I do with that? How do I compare rods and cones three microns to this?

Anybody remember anything about optics? So I have this big lens, right? And we have the eye on one side and we have the object on the other side. And we need to map some retina over here to some screen over here.

What's important to do-- what's the important thing to do in the map? Oh, come on. Do you all remember going to high school? No. OK.

So if you have a lens, rays go straight through a lens without being bent, right? That's one of the rules for lenses. So that is enough information actually to tell us the map. The map through a lens is so as to preserve angles. So if we figure out how closely spaced are the rods and cones on this side, that'll give me some angle that I can resolve.

And the question is whether that angle is bigger or smaller than the angle that's required to resolve the pixels. So the angle-- so if we make a small angle approximation, say that θ is on the order of $\sin \theta$, then the spacing between these is like three micrometers. The distance between the lens in your eye and the retina is on the order of, say, three centimeters,

something like that. So that's the angle at your eye.

And the question is, how does that compare to the angle at the screen? And so the screen, this is like three meters. But the pixels, there's 1,024 pixels in that range. And this distance is like 10 meters, something like that.

So the question is whether or not the angle subtended by the pixels is bigger or smaller than the angle subtended by the rods and cones, right? That's the issue. And so if you work that out, the angle between the rods and cones is on the order of 10^{-4} radian. And the angle between pixels is on the order of three times that.

So a couple of interesting things. You have enough rods and cones to see it, but only barely, by a factor of three, roughly. I'm not worrying about the fovea. The fovea has more. So this is just a crude approximation. I'm not worried about your eyeglasses.

But crudely speaking, you have enough rods and cones to resolve the pixels and another factor of three or so, which means, for example, that making a projector with three by three times more pixels makes sense. And making one that's got 100 times 100 times more pixels doesn't. And that's the kind of thing we'd like to work out when we're thinking about discrete representations for signals. How many samples do you need?

OK. So what we'd like to do today is figure out how sampling affects the information that's contained in a signal. We'd like to sample a signal-- so think about the blue signal here and think about the red samples. We'd like to sample a signal in a way that we preserve all of the information about that signal.

And as you can see from the example that I picked, it's not at all clear that you can do that. In fact, if you look at the bottom picture, I have coerced two signals to follow on the same samples. So the green signal is the $\cos 7\pi n/3$ and the red signal is the $\cos \pi n/3$. And they all go through the same blue samples. The same blue samples is shared by both of those signals.

So it's patently obvious that I cannot uniquely reconstruct a signal from the samples. That's absolutely clear. It's also clear by just thinking about the basic mathematics of signals.

A CT signal could move up and down arbitrarily between two samples. How could you possibly learn information about what happened between the samples by looking just at the samples? So it's not at all clear that you're going to be able to do this.

So let's take the opposite tact, which is to say, let's assume I only have the samples. What can I tell you about what the signal might have been? What's the relationship between the samples and the signal?

And the way to think about that-- one way is to think about something that we will call impulse reconstruction. If I only had the samples, what could I do to reproduce a CT signal? The simplest thing I might conceive of doing is replace every sample with some non-zero component of the CT representation. So I only have samples up here at nT , so generate a new CT signal that contains the information in the samples, that is to say x of n , which means I only have an integer number of non-zero elements in the CT representation.

So in order to make that signal have a non-zero integral, the things I represent each sample with better be an impulse, otherwise I would have finitely many non-zero points in a finite time interval, and the integral over the interval would be 0 always, right? So I have to use impulses.

So the simplest thing I could do would be to take every one of these samples and replace it by an impulse located at the right time, so put the n -th one at time t equals n cap T , put it where the sample came from, and scale the weight to be in proportion to the amplitude of x of n . That's kind of the simplest thing I could possibly do. That's called impulse reconstruction.

And then lets ask the question, how does the x that I started with relate to this x_P , this impulse reconstruction thing that I just made? And as you might imagine from the theory of lectures, that relationship is going to be simple, right?

So think about, what am I doing? I'm trying to think about I started with x of n -- I started with x of T , sorry. I turned that into samples. I turned that into x_P of t . And now I'm trying to compare those two signals. That's the game plan.

So think about x of P from the previous slide. It's weighted impulses shifted in time to the appropriate place. This x of n was derived by uniform sampling of x of t , so x of n was x of nT .

And since this impulse is 0 everywhere except where the argument is 0, it doesn't matter whether I call this thing nT or T , same thing, because the impulse only looks at t equals nT . So whether I call it nT or t is irrelevant. If I call it t , then this part has no n in it and I can factor it outside and I just get an impulse train. And not too surprisingly, x_P is just the product-- this signal is just this signal multiplied by an impulse train.

So if I derive xP by multiplying by an impulse train, your knee-jerk reaction is to say-- multiply by an impulse train?

AUDIENCE: Sounds like a Fourier transform.

DENNIS
FREEMAN: Sounds like a Fourier transform somewhere. If I multiply in time by an impulse train, what do I do in Fourier transform land? Convolve, right? Multiply in time, convolve in frequency.

So if this was my original x , this thing, if this is the transform of that thing, and if this is the transform of my impulse train-- transform of an impulse train in time is an impulse train in frequency. The impulses in frequency are separated by ω_s equal to $2\pi/T$. And the amplitude is equal to $\omega_s/2\pi$. So there's spacing and have an amplitude, both the spacing and the amplitude are $2\pi/T$, right?

And if I'm multiplying in time x times P , I convolve in frequency. So this is my answer. What's the relationship between x and xP ? It's multiplied in time by an impulse train, or it's convolved in frequency with an impulse train.

So the answer how does xP relate to x , it looks very similar for some frequencies. But there's a lot more frequencies. Not too surprising. I multiply by impulses. Impulses have all frequencies, right?

So what I've done when I've sampled it, if I think about the sampled signal being represented in CT as the multiplication by an infinite impulse train, I've introduced new frequency components to the Fourier representation. That's the main message. This slide is today's lecture. The way we can think about sampling in time is as convolution in frequency.

OK. So let's think about that a little more. What I just talked about was the relationship between the Fourier transforms of x and xP . But the goal is to think about the samples. So what's the relationship between the DT, the Discrete Time Fourier transform of the sampled signal, and the continuous time Fourier transform of this impulse reconstruction?

So I've already for you-- I've compared the frequency-- the Fourier representations of these two. As an exercise, you compare the frequency representations of those two. And figure out if any of these are the right way to look at it. So look at your neighbor.

So which one best describes the relationship between those Fourier representations? Number

1, 2, 3, or none of the above? OK. So 100%, I think.

So easier question, is x of e to the j ω -- x of e to the j ω , so this one, x of e to the j ω , is that a periodic or aperiodic function of ω ?

AUDIENCE: Periodic.

DENNIS
FREEMAN: Periodic. What's the period? What's the period of x of e to the j ω ? What's the period of e to the j ω ?

AUDIENCE: [INAUDIBLE].

DENNIS
FREEMAN: So I'm hearing about the three not quite correct answers. They all kind of have the right stuff in them. What's the period of e to the j ω ? What's the period of cosine? What's the period of cosine?

AUDIENCE: 2π .

DENNIS
FREEMAN: 2π . Thank you.

OK. So this one's periodic in 2π , so x of e to the j ω is periodic in 2π because e to the j ω is periodic in 2π , right?

How about xP of j ω ? Is that periodic or aperiodic? xP of j ω . xP is the signal that I got when I multiplied in the time domain x of t times p of t , p of t being an impulse train.

Take an impulse train times the time domain signal, and that's how I got xP . And then xP is the transform of that. Is the transform of xP periodic or aperiodic?

AUDIENCE: Periodic.

DENNIS
FREEMAN: Periodic. What's a period of xP ?

AUDIENCE: 2π over T .

DENNIS
FREEMAN: 2π over T precisely. So this one is periodic in ω . 2π -- I shouldn't write it that way, I should say that the period is 2π . And here, the period of ω is 2π over T .

So what's the relationship between Ω and ω ? In order to make a function-- in order to make a function that is similar, we're going to have to have ω is ω over T .

OK? We're going to have to convert the units of capital ω , which are radians, into the units of little ω , which is radians per second. So we need a time in the bottom. And if you want to be a little bit more formal about it, you can just write out the definitions. I did this last time. This was from two lectures ago.

So here's the definition of the discrete time Fourier transform, right? You take the samples and weight them by $e^{-j\Omega n}$. Here's the definition of the CT Fourier transform, where I've substituted x_P of t is this thing. It's a string of impulses, each weighted by x of n .

And then I interchanged the integral and the summation. And I don't worry about whether it's going to work or not. You have to take a following course, Course 18, in order to figure out whether that makes sense or not, but it does.

So I interchange the order and then the end part factors out and this just sifts out the value of ω -- the value of t and nT . So I replace then this t with nT . The integral goes away. And I end up with something that looks almost exactly like that, except that capital ω has turned into a little ω times T .

So the discrete time Fourier transform is just a scaled in frequency version of this impulse-sampled original signal. So the impulse reconstruction is related to the Fourier-- the Fourier transform of the impulse reconstruction is related to the Fourier transform in samples by scaling frequencies that way.

So those representations have the same information precisely, except for the scaling of frequency. The period in the bottom waveform is 2π . The period in this one is 2π over capital T .

So the answer to the question then is easy. The original question was, under what conditions can I sample in a way that preserves the information in the original signal? Well, this diagram makes it relatively clear. If the original x was just one of these triangles and x_P is the periodic extension of that, then so long as the periodic extensions don't overlap with each other, I can derive x from x_P by simply low-pass filtering. Throw away the frequencies that got introduced by the convolution with an impulse train.

So as long as the frequencies don't overlap, as long as there's a clean spot here where there's

nothing happening, as long as the frequencies of the periodic extensions don't overlap, then I can sample in a way that contains all of the information of the original, so long as when I'm all done, I low-pass filter. That's called the sampling theorem.

So the sampling theorem, which is not at all obvious if you don't think about the Fourier space - if you only started with the time domain representations that I showed in the first few slides, it's not at all obvious that there even is a way to sample in an information-preserving fashion. But what we've just seen is that it's really simple to think about it in the Fourier domain. Thinking about it in the Fourier domain gives rise to what we call the sampling theorem, which says that if a signal is bandlimited-- that has to do with this overlap part-- if the signal is bandlimited that means that all of the non-zero frequency elements are in some band of frequencies, nothing outside that band.

Outside some band, I don't care what the band is, but outside that band, the Fourier transform has to be 0. If the Fourier transform is 0 outside some band, then it's possible to sample in a way that preserves all the information so long as I sample fast enough. So if the original signal is bandlimited so that the Fourier transform is 0 for frequencies above some frequency ω_m , then x is uniquely determined by its samples. Uniquely means that I can do an inverse.

So it's uniquely determined by the samples if and only if ω_s , this is sampling frequency, which is $2\pi/T$ -- that's the period of the impulse train. So if $2\pi/T$ exceeds twice the frequency of the band limit. We'll see in a minute where the factor of 2 comes from. The factor of 2 comes from negative frequencies.

So the sampling theorem says that there is a way for a certain kind of signal. Signals that are bandlimited can be sampled in a way that preserves all the information.

So here is a summary. If you sample uniformly-- that's not the only kind of sampling we do in practice, but it's the basis of all of our theories for the way sampling works, so that's the only one we're do in 003. But just for your intellectual edification, there are more sophisticated ways to sample. And in fact, that's a topic of current research.

But for the time being, we're only going to worry about uniform sampling. If you sample a signal uniformly in time, that is every capital T seconds, then you can do bandlimited reconstruction, which means that replace every sample with an impulse weighted by the sample weight and spaced in time where it would have come from. Then run it through an

ideal low-pass filter to get rid of the stuff that high frequency copies. And what comes out will be equal to that, as long as you've satisfied this relationship, that the frequencies that are contained in the signal must be less than the sampling frequency over 2.

OK. So what's the implication of the sampling theorem? Think about a particular problem. We can hear sounds with frequency components that range from 20 hertz to 20 kilohertz. What's the minimum-- what's the biggest sampling interval T that we can use to retain all of the information in a signal that we can hear?

So what's the answer? 1, 2, 3, 4, 5, 6? OK. A sort of shrinking number of votes and sort of a shrinking number of correct votes. So it's about 75%.

The only tricky part really is thinking about frequencies. So I told you frequencies in the commonly used engineering terms-- I told you that we hear frequencies from 20 to 20,000 hertz. In this course, we usually think about frequencies as omega radian frequency. Hertz are cycles per second.

There's 2π radians-- hertz are cycles per second. Radian frequency is cycles per second radians per second. There's 2π radians per cycle. OK. Sorry. I screwed that up.

Hertz is cycles per second. Omega is radians per second. The conversion is 2π radians per cycle.

So we need to have-- the highest frequency in the signal has to be smaller than the sampling frequency divided by 2. The highest frequency in the signal is $2\pi f_m$, if f is frequencies 20 to 20 kilohertz. And the sampling frequency is 2π over capital T . So you clear the fraction and figure out that T has to be smaller than 25 microseconds.

OK. So the idea then is that there's a way of thinking about any signal with a finite bandwidth-- there's a way of sampling using uniform sampling so that you can sample that signal without loss of information. All you need to do is figure out how frequently you need to sample it. Yes?

AUDIENCE: Implications? A signal is not bandlimited.

DENNIS If a signal is not bandlimited, you have a problem.

FREEMAN:

AUDIENCE: Is it possible to [INAUDIBLE]?

DENNIS
FREEMAN:

So the question is, to what extent-- is it possible to sample a signal that is not bandlimited? So that's the next topic.

So the question I'm going to address now is, well, what happens if you don't satisfy the sampling theorem? What if there are frequency components that are outside the admissible band-- the admissible region of frequencies?

So to think about that, I'm going to think about this as a model of sampling. So I think about-- and the value of the model is that it's entirely in CT. It gets confusing when you mix domains and you have to compare this kind of a frequency to that kind of frequency.

So what I'm going to do is think about it entirely in CT by making a model of how sampling works. Sampling is equivalent to take the original signal that you're trying to sample, multiply by an impulse train showed here. You get out this x_p thing that we've been talking about. And then run that through an ideal low-pass filter. And if this comes out looking like that, then the sampling preserves all of the information.

Now what I want to do is think about what happens when I put in a signal for which that doesn't hold. So let's start with the simplest possible case. Let's think about a tone, so a signal that contains a single frequency components. So I'm representing that by a cosine wave, $\cos(\omega_0 t)$. So I'm representing that by two impulses, Euler's equation.

And I'm thinking about some sampling waveform, where I'm looking at it here in frequency, so the spacing is $2\pi/T$. And if the frequencies in the signal are smaller than $1/2$ of the maximum frequency ω_s , everything should work. And you can sort of see in the Fourier picture what's happening.

When I convolve these two signals-- I multiply in time, so I'm convolving in frequency. When I convolve this with the impulse train, this impulse brings this pair of frequencies here. But this impulse brings this pair up here. And then this one brings this pair up here.

So I repeat. So I get a repetition then of the original two at integer multiples of ω_s . OK. So if I low-pass filter then with the red line, everything's fine. I end up with a signal with the output that's the same single as the input, no problem.

What happens, however, as I increase frequency? Same thing. The originals are reproduced here. The modulation by this brings this up to here. The low-pass filter still separates it out.

The problem is here. Now I'm running into trouble, because the original signal fell right on the edge of my limit, ω_s over 2. And the problem's even more clear if I go to an even higher frequency. So now, this component is coming out here, which is outside the box.

The thing that's happening, though, is that convolution with this guy is bringing in an element that is spaced by ω_s and is now inside the box. That's bad. Everybody see what's happening? So I'm just studying what happens as I have one frequency and vary the frequency-- if I have a tone at a single frequency and I vary the frequency of the tone, as long as the frequency is inside my limit, ω_s over 2, I'm fine. The low-pass filter reconstructs the original.

But something bizarre happens whenever I'm outside. And to see what's going on, it's easiest to see what's going on by making a map between the input frequency and the apparent frequency, the apparent frequency of the output.

So what happens if I'm at a low frequency, I'm on a linear relationship between the input frequency and the output. The output reproduces the input exactly. And you can see as I'm increasing the frequency, I'm just sampling this function at a different place. But when I go higher, it appears as though the frequency got smaller.

We call that aliasing. This is a question to get you to think through aliasing. But in the interest of time, because I have a demo, I'll leave this for you to think about. So the question is, thinking about what's the effect of aliasing and where do the new frequencies land relative to the input frequency-- they have this funny folding property. And the effect of the folding property kind of wreaks havoc.

So I'll skip this for now and just jump to the idea that the intuition that we get by thinking about single frequencies carries over to complex frequency representations. So now what if my message, what if my input had this triangular frequency Fourier transform, rather than just a single spike? And so I'm sampling it as showed here, so this is my impulse train and frequency.

And as before, the message gets reproduced at integer multiples of ω_s . And as long as the bandwidth of the message is small enough, I can put the red low-pass filter to eliminate the copies from the periodic extension, and I get an output that's equal to the input.

The problem is that if I increase the bandwidth of the input, as I increase the bandwidth of the

input, the margin between the base signal and the periodic extension gets smaller. And if the bandwidth gets too great, they begin to overlap. So now if I low-pass filter this signal, I pick up the desired blue part of this, but the undesired green part of that. And the result is that the signal is not a faithful reproduction of what I started with.

Same sort of thing happens if I hold the message constant and change the frequency spacing. As long as I sample with big enough frequencies-- that is small enough times. There's an inverse relationship between frequency and time. If the frequency is big enough so the times are short enough, I can reproduce what the signal looked like.

But now if I change the sampling to happen slower, I start to get aliasing. So it's exactly analogous. So that means that if you have a fixed signal, there's some minimum rate at which you have to sample it in order to not lose information.

So what I want to do now is demonstrate that by thinking about music. So I have a cut of music that was originally taken from a CD, so it was sampled at 44.1 kilohertz. That's standard CD frequency sampling.

And I've re-sampled it at 1/2, 1/4, 1/8, 1/16. And I'm going to play what happens when I do these various re-samplings.

[MUSIC PLAYING - JOHANN SEBASTIAN BACH, "SONATA NO. 1"]

[MUSIC PLAYING - JOHANN SEBASTIAN BACH, "SONATA NO. 1"]

[MUSIC PLAYING - JOHANN SEBASTIAN BACH, "SONATA NO. 1"]

[MUSIC PLAYING - JOHANN SEBASTIAN BACH, "SONATA NO. 1"]

[MUSIC PLAYING - JOHANN SEBASTIAN BACH, "SONATA NO. 1"]

OK. Some you may have been able to tell the difference. So the idea is that by decreasing the rate at which I'm sampling it in time, I get fewer samples in the total signal, which makes it easier to store and increases the amount of stuff that I could put on a given medium. That's good.

Problem is, it doesn't sound as good. So every step down this path resulted in 1/2 the

information of the previous one, which means that I could double the capacity of your MP3 player. But there were distortions added because of this aliasing problem, because some of the frequencies at the lower sampling rates were too large to be faithfully reproduced by the sampling. They got moved, they got aliased to the wrong place. That gives rise to sounds that are inharmonic and they sound bad.

So this just recapitulates what we were seeing in the demo. I started out barely having enough bandwidth to represent the original signal. And as I made the sampling frequency smaller, making the distance between samples bigger-- as I made the frequency of the sampling smaller, I started to get overlap, and that's what sounded funny. So the slower I sampled, the more overlap and the funnier it sounded.

So the question is, what can you-- how can you deal with that? One way you can deal with that is what we call anti-aliasing. So it's very bad if you put a frequency into a sampling system where the frequency is too big to be faithfully reproduced, because it comes out at a different frequency that cannot be determined from the output alone.

So how can you deal with that? One way you can deal with it is to pre-filter, take out everything that could be offensive before you sample. That's called anti-aliasing. And the result is not faithful reproduction of the original. But it's at least faithful reproduction of the part of the band that is reproduced.

So it's not distortion-free. The transformation from the very input to the very output is not-- there's not a unity transformation, because you violated the sampling theorem. But at least you don't alias frequencies to the wrong place.

So the result is-- I'll play without anti-aliasing, with anti-aliasing, without, with, without, with.

[MUSIC PLAYING - JOHANN SEBASTIAN BACH, "SONATA NO. 1"]

OK. So the final one didn't sound exactly like the original, but at least it wasn't grating. So the idea, then, is just that sampling's very important. And by thinking about it in the Fourier domain, we get a lot of insights that we wouldn't have gotten otherwise. See you tomorrow.