

SPEAKER: 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 to view additional material from hundreds of MIT courses, visit MIT OpenCourseWare at ocw.mit.edu.

PROFESSOR: OK, we talked about this a little bit last time. We were talking about the detection of vectors in white Gaussian noise. When we talk about vectors we'll often refer to white Gaussian noise as noise where each component of the vector is independent of each other when they're all the same variance. Usually we take the variance to be $N_0/2$, capital $N_0/2$. And we'll say more about that as we go on, but that's just-- I guess one thing I ought to say about it now-- people keep wondering why we call this $N_0/2$ instead of something else. As I said before, there's really no reason for it except custom.

The one important thing that you can always remember and which is always true is that any time you're talking about a sequence of real noise variables they all have variance $N_0/2$, and the same coordinate system that you're using to measure the signals. OK, the only thing that ever appears in any of these formulas is a ratio of signal power or signal energy to noise signal power. And if we're up at passband, we're dealing with the power, which is two times larger than that at baseband. And because of that-- and this is what really gets confusing-- is that when you talk about $N_0/2$ at passband, you are talking about something which is twice as big as the $N_0/2$, the same $N_0/2$ you were talking about at baseband. And the reason is since the signal is twice as big there, the noise is also said to be twice as big. I can't do anything about that. It's just the way that everybody does things.

The other thing that everybody does-- since everyone gets confused about that-- is after they get all done dealing with anything in a paper they're writing or something, they always look at the signal to noise ratio that they have and they remove all the factors of two that they know shouldn't be there. So that you shouldn't trust anything in the literature too much as far as factors of two are concerned. And I try to be careful in the notes about that, but you shouldn't trust the notes too far along those

lines, either. So well eventually I'll get the notes straightened out on all of that but I think they're pretty close now.

But anyway, we were looking at this question of how do you detect antipodal vectors in white Gaussian noise? And the picture that we can draw is this. Namely we have two signals. One is the vector, a , one is the vector, minus a . It's in some finite dimensional system, but we're viewing it as far as drawing a picture as a two dimensional system. So a has some arbitrary component in the first direction. Some arbitrary component in the second direction. Minus a is, of course, the reverse of that. This point right in here is the zero point, which is halfway in between them. And the output that we observe is either plus or minus a , plus this independent zero mean noise, Z , which has the kind of circular symmetry indicated here with these little circles. Each of the Z sub i have the same variance, they're independent of each other. And when you write down the likelihood of the probability density of this output, given that the hypothesis was zero. Namely that plus a was the signal which was chosen.

OK, remember in all of these things there's this process now going on that we usually don't talk about anymore. But there's an input coming into the communication channel which we're now calling capital H . It's the hypothesis-- which is the thing you're trying to detect when you're all done-- that input which is one up to capital N , or sometimes zero up to capital N minus 1. Is then mapped into a signal from a single set of capital N different signals. So they're impossible signals in this signal alphabet. You map the hypothesis into one of those. From those, you generally form a waveform. This waveform might be modulated up to high frequency, back to low frequency again. Detected or whatever. You got some vector, v , at that point. Which is a sequence of samples that you're going to be taking as far as most cases are concerned. We'll talk more about that later today. But anyway, v is a vector which is plus or minus a at this point, plus this Gaussian noise.

We can write down the probability density of that vector, v , which is if hypothesis zero occurs. Namely if a zero enters the communication channel, plus a is the signal

which is chosen. Then what happens is that the output is a plus Z. Which means that the probability density of the noise is v minus a . So we have this probability density here. When we look at the log likelihood ratio, we're taking the logarithm of this probability density divided by the probability density of the alternative hypothesis. Namely v given one. Which is the same as this formula except there's the plus a there instead of a minus a . So you get this thing here.

Now, why I wanted to talk about this today is we're going to talk about the complex case also and something very, very peculiar and funny happens there. OK so the log likelihood ratio is the scaled difference of the energy of the distance between v and a . Which is this term here. This is just a squared distance between the vector, v , and the vector, a . It's v minus a is that distance there, squared. And the other term here is the term that comes from the probability density of v given one. Which turns out to be that distance there squared. So you have the difference between these two things. This is just the inner product of v minus a with v minus a . And if you multiply that all out it's the inner product v with itself, plus the inner product of a with itself, minus the inner product of v and a , minus the inner product of a with v .

The only things that don't cancel out between these two things is the inner product of v with a , the inner product of a with v , the inner product of v with a , the inner product of a with v . So those things last because there's a minus sign here and a plus sign here. There's a minus sign here and a plus sign here. So the plus and minus signs cancel out, so you just get four of these terms here, which is four times the inner product over n_0 . What happens to that geometrically? What is the inner product of v and a ? Well it's the projection of v on the vector a . Which happens to be the line between minus a and plus a . Mainly the fact that it's the line between minus a and plus a is the thing which is valuable whether you're dealing with antipodal communication or any other kind of communication. You're always looking at this line between two points.

So what this thing says is you form the inner product, which says drop a perpendicular from Z down to here, and in terms of where that perpendicular lands here, you make your decision. Namely you compare that with the threshold. OK? So

we have two different ways of doing this. One of them is compare this distance with this distance. Or actually here you compare the square distance here with the square distance there. Which says, if you're using maximum likelihood then the threshold that you're dealing with here is zero and you simply make your decision on whether the log likelihood ratio is positive or minus. Which means in terms of the projection theorem here, what you're doing is taking a perpendicular bisector of the line between minus a and plus a , and putting a plane there and that's the plane that separates what goes into one and what goes into zero. This goes into zero. This goes into one.

OK, so is it clear to all of you that this is really saying the same thing as this? This inner product just comes from here. You can either look at this as minimum distance decoding. You just find the point which is closest to what you receive. You find the hypothesized signal, which is closest to the actual observation that you make. You make your decision in terms of that. Or you do this projection and make your decision in terms of that. And if you use a triangle thing which says that this squared distance plus this squared distance is equal to that squared distance. We all remember that from third grade or something. I don't know when. But that simply says the same thing that this says. This just says it more generally. In terms of an arbitrary finite conventional vector, rather than just the case of where you're looking at two dimensions.

OK that's-- we will probably come back to look at that in a little bit, but now I want to look at complex antipodal vectors in white Gaussian noise. So the set up there is that the input is some vector. I usually use u 's to mean complex numbers. So the vector there is u_1 up to u_j where u_j is a complex number. So we're dealing with complex vectors at this point. So we have two points, minus u and plus u . Where instead of being a real vector they're complex vectors. And if you can't visualize things geometrically, in terms of complex vectors, join the crew. I can't either. The only thing I can never do is talk about real vectors, try to get some idea of what's going on from that, and use mathematics for the complex vectors.

Because the reason we use complex vectors is that, analytically they're just as

simple as real vectors. The reason we use real vectors is because we can draw pictures of them. I defy anybody to draw a picture in four dimensions. Some books will do it, but I can't understand them. And anyway. OK, so under hypothesis zero what gets sent as u ? And under hypothesis one, what gets sent as minus u ? So we're still talking about binary communication and if you're talking about antipodal vectors, you can't do much other than talk about binary communication. Because if you're sending a , the only vector antipodal to a is minus a . And then you're stuck and you're done. So we're still talking about binary vectors. Remember the reason why we're doing this-- because one of the things we did last time, one of the things that's in the notes and stressed again and again is that once you understand the antipodal case, you can translate those two points anywhere you want to and the maximum likelihood decision and the MAP decision are still the same thing.

You take those two points and you just translate them in space until the mean between them sits on the zero point. And then you're back to the antipodal case again. OK, so the reason why we're doing this is really so we can look at the more general case. But we don't have to have that mean sitting around all the time. OK, Z then is going to be a vector of j complex IID Gaussian random variables. IID real and imaginary parts. Namely the real part of each Gaussian, complex Gaussian random variable has variance $n_0/2$ and the imaginary part also has variance $n_0/2$. These complex vectors, if you look at the probability density for them and you draw it in two dimensions, one for the real part one for the imaginary part, you get this circular symmetry that we've always associated. Those are supposed to be circles and not ellipses. And those are sometimes called proper complex Gaussian random variables. Because almost everywhere where you see complex Gaussian random variables, the real and imaginary parts are independent of each other and both have the same variance. Again, when you look at formulas in papers, formulas everywhere else, they are almost always assuming this kind of circular symmetry. Or what's often called proper complex random variables. Sort of accepting the fact that anything else is very, very improper. It's improper because formulas don't work in that case.

OK, so we have a vector of these complex IID Gaussian random variables. Under H

equals zero, the observation, v , is given by v equals u plus Z . And under hypothesis one, the observation is given by minus u plus Z . So I have exactly the same cases as we had before. OK in other words almost all formulas stay the same when you go from real to complex. But I don't trust the complex case and you shouldn't either until you at least go through it once. So what I'm going to do now is translate this complex case to the real case. In other words, for each complex variable I'm going to make two real variables. One the real part and one the imaginary part. We know that the real part and the imaginary part are independent of each other. And if these Gaussian random variables are independent of each other, then the real and imaginary parts of each of them are independent of the real and imaginary parts of each of the other side. OK?

So we can go from j Gaussian random variables, independent Gaussian random variables, which are complex. To $2j$ Gaussian random variables, which at this point are going to be real. OK so it's just a translation from j complex variables to $2j$ real variables. Again we can't draw pictures of things in this j dimension, we can start to draw pictures in $2j$ dimensions. If you talk about a probability density for a complex random variable, what are you talking about? How do you write the probability density for just a plane Gaussian complex random variable? What is it? Anybody know what it is? Is it one dimensional or is it two dimensional? What does probability density mean? It means probability per unit area. What does area mean when you're talking about complex numbers? Well you sort of mean what you've drawn there, yes. And you're looking at areas in this complex plane here. So that in fact when you write the probability density for a complex random variable, what you have already done whether you want to do it or not is you've converted the problem to real and imaginary part. That's what the probability densities are.

Excuse me for a belaboring this but, if I don't belabor it I mean there's a catch here that comes along in a little bit. And you won't understand to catch if you don't understand why these things are almost the same as real variables up until the catch comes. OK, so we're going to deal with these $2j$ dimensional real vectors. The components will be real part of u_j , imaginary part of u_j for what goes into the

channel. And we'll let capital Y capital Z prime be the two j dimensional real versions of V and Z .

OK so that we'll call Y the real part, an imaginary part of Z . And you notice that what's going on here is the same thing that's going on when you modulate QAM. You take a complex signal, you multiply it by either the $2\pi j$ carrier frequency times t , and then we started to look at orthonormal expansions, you remember that what we looked at-- as far as the real signals that were actually being transmitted on the channel-- was the real part of that u of t times z to the blah, blah, blah, and the imaginary part of u of t times blah, blah, blah. So you've got two orthonormal functions in place of one complex orthonormal function. And that's the same thing that's going on here. It's just not with modulation put in, it's just dealing with the real parts and imaginary parts directly.

OK, so if we do that we get a bunch of equations. They're sort of familiar looking equations by now I hope. This is just the probability density of this real $2j$ dimensional random variable. Which is all this junk that we're used to seeing. We can collapse that into e to the minus the norm squared of y minus a . This is the norm squared in this $2j$ dimensional real space. It's not the norm squared in this complex space. What gets confusing is that those two norms are exactly the same. As we'll see in just a few minutes. But anyway, what we're dealing with now is this norm in real space.

OK, note that's y -- oh let me translate this one for you. If we think of this v that we received j complex random variables as being: real part of v_1 , imaginary part of v_1 ; real part of v_2 , imaginary part of v_2 ; and so forth, then $y_{2j}^2 - 1 - a_{2j}^2 - 1$. I can't ever get these formulas right. That should be $a_{2j}^2 - 1$. There. This squared, plus this squared-- OK, in other words the real part squared of the difference plus the imaginary part squared of the difference-- is really just the same as $v_j - u_j$ squared.

OK, in other words you take the complex number $v_{sub j}$, you subtract off the real number $u_{sub j}$. And the way to do that, you visualize this one complex variable in

the complex plane, and what you're doing is you're taking the square of the real part of the distance, you're adding it to the square of the imaginary part of the distance. OK, all of this is stuff you learned in high school. Just viewed in a slightly different way. OK, now when you take the probability density with respect to these complex vectors-- which is what I want to get at-- probability density of these complex variables really means the same thing as that with the real variables. But then you wind up with the magnitude of v_j minus u_j squared. And this term is really exactly the same as these two terms there. So when you take this probability density, you wind up with these terms the same and with these terms the same. OK, in other words the complex norm squared is the same as the real norm squared, when you go from complex to real and imaginary parts of, well-- as I said, this is the same as that.

OK so when we look at the log likelihood ratio, then, let's do the log likelihood ratio in terms of the real parts first. We get the difference between this norm squared of y minus a , and the norm squared of y plus a . This is the part that comes from the hypothesis zero, this is the part that comes from the hypothesis one. OK, so we get these two terms here. When we take the inner products here, we get the same thing that we got before. Four times the inner product of y and a . Now, the whole reason for going through all of this is this next formula. When you do this, you wind up with this very bizarre four times the real part of the inner product of v and u , divided by N_0 .

And you get it in exactly the same way that we got it before. Namely you take this norm here, which is an inner product squared of v minus u . Let me write it out. I'll write it out here. Norm squared of y minus a , is the norm squared of y plus the norm squared of a , plus the inner product of y and a , plus the inner product of y and a . What's the sum of these two inner products? This inner product, by definition in terms of integrals, is the integral of y times a -- complex conjugate. This is a times y -- complex conjugate.

So in one case the complex conjugate is here. In the other case it's on the other term. In other words this and this are complex conjugates of each other. What happens when you add two complex conjugates? You get the real part of the two of

them. Okay so when you add these two things you just get that real part there. OK, and then when you do the other term, the same thing happens. The same cancellation that we had before occurs. And these two inner products add up so you wind up with the real part of v over N_0

When we look at the picture here, what does it mean? Well I suggest you first look at the one dimensional case here. Namely on this one dimensional case think of V as being a one dimensional complex random variable. Then we can draw a picture. The picture make sense. And what we're dealing with is the real and imaginary parts, and these distances here, when you talk about the norm of V minus a -- namely what corresponds to this line here, the length of this line-- what do you really mean by it? If you took the inner product, if you took the norm of v , with i times a -- namely that the square root of minus 1 times a -- would you get the same thing or wouldn't you?

If I take a complex number, and I take the inner product of that complex number-- namely the product-- of that, when I take the inner product of y , is this the same as the inner product of y and i times a ? Not at all. The two things are totally different. Namely, inner products are complex things. Norms are real things. And these norms, when you're dealing with complex numbers, have real parts in them. In other words, this distance that we're talking about here is not just the norm squared-- well it is the norm squared-- because the norm has this complex feature built into it. Because people kept making that mistake all the time. So they fudged the mathematics to make it come out right. But after doing that, you have to fudge the mathematics to come back to something that makes sense here. So you have the real part of this inner product, here.

So in fact, what you're doing when you're taking the inner product of two vectors and you're trying to relate it to this plane here, this separation plane, is you have to look at that separation plane. You have to look at that projection in terms of real numbers. Namely, you have to look at the projection. First as being a complex projection of v onto a , which gives you a complex number. And then after you do that you have to you visualize yourself in a two dimensional real space. And you

have to project once more from the two dimensional thing to the one dimensional thing.

And here where we're just looking at one dimension to start with, we have to draw it as a two dimensional space. And suddenly we are dealing with this real part there while we're not dealing with that here. Which is why when people say minimum distance detection when they're dealing with complex numbers it sounds very, very simple. But in fact, it's not so simple. If you view this in the complex plane, is this a linear operation or isn't it? When you're looking at things as complex vectors. Is this thing a sub space of the complex vector space? No, it's not. It's not a sub space. Because, to be a sub space you have to be able to multiply by arbitrary scalars-- which includes complex numbers-- and stay in the same space. And here the complex numbers are important.

OK? You should go back and think about that. You will be confused about it for the first ten times you think about it. For those of you who stick with it and carry through on it, you'll be happy because you'll never be confused about it again. OK, anyway that's real numbers there. And the most straightforward way to deal with complex noise is to first turn it into real noise. If you do that you never get confused. And otherwise you only have half the analytical work. You only have half the writing to do. But you never know whether you've done the right thing until you go back and check. OK the probability of error for maximum likelihood detection, in other words where the threshold for the log likelihood ratio is zero, is simply the same thing that it was before. Namely it's the Q function. This tale of the Gaussian normal function. Of the norm of a divided by the square root of N_0 over 2. In other words it's the length of a divided by the length of a one standard deviation of the noise.

When you put that in terms of, well, if you write this as the square root of the norm squared then you get this formula here. Because e^{-b} is just the energy of these antipodal signals. When you look at this in terms of the complex random variables, you get the same thing. OK? You get u instead of a because, in fact, in the complex plane and the real plane, distances turn out to be the same. But again, in all cases it's square root of $2eb$ over n_0 Now, that is true for any vector at all where these

norms are appropriate. When we start dealing with functions what happens? When we start dealing with functions, what we're going to do is we're going to take this vector, we're going to turn it into a wave form. We're going to transmit the wave form. The wave form is going to come back to us. We're going to demodulate it, get back to a number again. And that's the next thing that I want to talk about. Because what I want to convince you of is the property of white Gaussian noise which is so important. Is that it doesn't make any difference how you modulate. All modulation systems are the same. All modulation schemes are the same. All frequencies are the same. There is no way you can avoid white Gaussian noise. There is no way you can get screwed by it. No matter what you do, the same thing happens.

You can always take all these problems where you're dealing with wave forms. You can convert them to finite dimensional vector problems. And when you convert it into a finite dimensional vector problem all of the orthonormal functions that you're using all pass away. Because none of them are relevant anymore. OK? That's the bottom line of all of that. OK.

We haven't really talked about M-ARY hypothesis testing. So I want to talk about it a little bit now. I talked about it just a shade, but not much. When we want to detect between m different hypothesis-- namely in the vector case we're going to now be detecting not between antipodal signals but m signals which are placed any place at all. We already said what the MAP, optimal MAP test was. You see an observation, you're trying to guess what the input was, or what the hypothesis was. And in general, the MAP test says try to find that j , namely that hypothesis, for which the a priori probability of hypothesis j , times the likelihood-- namely the probability that you see v given h of v , given j is maximum. OK, this is just standard formula for finding a posteriori probabilities. Where you factor out the marginal on, when you cancel out the marginal on v .

In other words, what this rule says is to do MAP testing, what you do is you find the a posteriori probabilities of each of the hypotheses and you choose the a posteriori probability which is largest. Perfect common sense. The way we're going to do that, the way which is particularly convenient, is you do it the same way that we've been

doing all along for binary hypotheses. The way to do a MAP test, at least one way to do a MAP test, is you compare every pair of hypothesis and you choose the most likely of each pair. And when you've got it all done, you have a winner. OK?

Mainly you can always compare any objects if they're comparable. And after you compare each pair, you take the one which beats every other one and that's the winner. OK? So what you do is you do a pairwise test between each hypothesis. Namely, the likelihood ratio of m relative to m' . Is the likelihood of the output conditional on m , divided by the likelihood of the output conditional on m' . You compare it with the a priori probabilities, and the point here is that nothing really has been added. You have the same problem you had before, OK? Nothing new. It's just gotten n square times as complicated. The computation is free now, so you have exactly the same problem that you had before. If you have to write it out on paper, yeah it's much more complicated. But conceptually, it's not.

You have to remember that the signals are not antipodal here. But what we're dealing with mostly at this point is this Gaussian noise case. And here, what you observe, is signal plus noise. And Z is zero-mean jointly Gauss and s is discrete with n possible values. OK, so let's see what that means. Here's a picture of it. If you have three signals which are each two dimensional vectors, suppose one of them is s_0 , suppose one of them is s_1 , suppose one of them s_2 . OK? And now you want to pairwise, see which one is most likely. And let's think of doing this first for the maximum likelihood case. What do you do? You set up a perpendicular bisector between s_0 and s_1 . That's this line here. And if you weren't to worry about s_2 , everything on this side would go into H equals zero. And everything on this side would go into H equals one.

Namely, whatever's closest to this point gets mapped into it. Whatever's closest to this point gets mapped into it. If you're doing MAP testing, what happens? In the test between this and this you had the same orientation for this line, but it just gets shifted a little bit this way or a little bit this way. OK?

Then you compare this with this and you get this line here. Same argument as

before. It's just comparing two things are not antipodal they've just been shifted off from the origin a little bit. But for the maximum likelihood you still take the perpendicular bisector between them. And then you compare these two and you got a perpendicular bisector between those. And these perpendicular bisectors, in two dimensions, always come together at one point. And I don't know why. And if you looked at it often enough, you probably know why. And you could probably prove it in about ten minutes. But in fact these things always come together somehow. If you do the MAP test, they always come together also. You can shift each of them in arbitrary ways and somehow they always come together in this point.

OK, the separators between decision regions here are the set of points where the real part of the inner product, $\text{Re}\{vu\}$, is constant. OK? Again, for dealing with complex vectors, you got to both do the projection and then do the projection again onto the real part of this projection. So it's sort of a two way projection. Because probability densities in j dimensional complex space are really $2j$ dimensional quantities. And when you're comparing them, you really have to compare things in terms of that $2j$ dimensional probability density. OK so that's why that comes out. These are best visualized in separate, real, and imaginary coordinates. And for maximum likelihood detection, the regions are Voronoi regions. OK? We talked about Voronoi regions in terms of doing quantization.

And we found out if you wanted to minimize the mean square error, what you did was you set up regions, which are perpendicular bisectors between all the points. And here you get the same perpendicular bisectors between the points. And everybody-- because of that-- thinks that quantization has a great deal to do with error probability when you have large sets of signals. And it probably has something to do with it, but I don't know what other than the fact that you've got Voronoi regions in each case. Which is what you get.

OK, so that's where we are with both complex vectors and real vectors. I want to now just restrict attention to real wave forms so I don't have to keep going back and forth between the real and imaginary case. If you're thinking in terms of QAM, we're now thinking in terms of what goes on at passband. Why do we want to think of

what goes on at passband? Because that's where the noise hits us. And in a fundamental sense, all of the stuff about QAM really isn't fundamental.

I mean, it's all done-- all this stuff down at passband-- is all done because people thought it was easier to implement things there. It's the only reason for all of that mess with dealing with all of these complex signals. If we really want to deal with the problem in a fundamental way, what we want to do is to choose a signal set up at passband. Do detection up at passband. And then after we find out what the optimal detection is up at passband, see if we can actually implement that down at baseband. So the fundamental problem is looking at single sets up at passband and analyze what they all mean.

OK, so we're going to generalize both PAM and QAM. And now we're going to look at the general problem where what we're dealing with is a single set. Which is m signals. Each of them we're going to visualize as a vector in j dimensional space. m different signals, j dimensional space. Don't confuse the dimension of space with the number of signals. OK? You can have an arbitrarily large dimensional space and just binary signals. Or you can have an arbitrarily large set of signals and you can be dealing with it. In PAM, for example, it's just all done in one dimension. So j there is equal to 1. QAM j is equal to 2. We now want to look at more general things. Partly because we want to look at orthogonal wave forms.

We want to look at orthogonal wave forms for two reasons. One is that we would like to show that by using orthogonal wave forms, you can reach what we've called the capacity of a white Gaussian noise channel. And two, when we get to studying wireless it's very, very useful to base signal sets on orthonormal sets of functions. And we'll see why each of those things happen as we move on.

OK so we're going to denote the single set as the set of vectors, a_1 up to a_m . And in that signal set, we will denote the vector, a_m , as a j dimensional vector, a_{m1} , a_{m2} , up to a_{mj} . So j is at the dimension. m is just a component of these vectors. I'm going to create a set of capital J orthonormal wave forms. They can be anything at all. I don't care what they are. I'm going to use those

orthonormal wave forms in order to modulate the signal-- which is now a vector-- up to some waveform. This is really the standard way we've been turning signals into waveforms all along. It's just that now we're looking at the general case instead of all of these specific cases that we've been looking at. All of the special cases all fit into this same category.

So we have these capital M different waveforms. And we're going to transmit one of them and then at the receiver we're going to try to decide which one was sent. OK, well one of the reasons why I'm going through all of this generality is that there's an issue we haven't talked about yet. All of the stuff we've done on detection so far we have had one hypothesis. Could be M-ary could be binary. We have sent something. We have received something. We have made a detection. OK? In other words, for all of the antipodal stuff we've done, we built a communication system. We set it all up. We transmitted one bit. We've received the one bit. We've made a decision on it. Then we've torn down the communication system and we've gone home.

You really want to transmit a whole sequence of symbols or signals or waveforms. So we want to deal with that now. So we need some way to transmit a succession of M-ary signals. And we'll call this succession of signals-- mainly the signals are the things that get chosen from the signal set-- we'll call them x_k , k of z . Which is what we've been calling them all along. We've been transmitting a sequence of things when we're dealing with PAM. And aa in am. Why do I call them x_k instead of a_k ? Well I can't call them a_k because when I talk about a_k I'm talking about the k 'th signal in the signal set. And here what I'm talking about now is transmitting a sequence of choices. Each one of these choices, the first choice is a choice from this set here. The second thing that I send is a choice from this set. The third thing that I send is a choice from this set. So x_1 , x_2 , x_3 , and x_4 and so forth are different choices among these M-ary signals. If m is 2 to the 6-- OK in other words, every time I transmit a signal I'm transmitting six bits.

OK. In a communication system we transmit six bits. Then we transmit another six bits. Then we transmit another six bits, and so on forever. OK, so I need to talk

about these things as ways of a succession of signals. The thing that I'm trying to get at is, how do you know when you send one of these signals that you don't have to worry about the other signals? How do you know that they don't interfere with each other? Well, we sort of solved the problem of them interfering with each other in dealing with Nyquist, but we haven't dealt with that problem at all since we started to talk about random processes. So we don't know whether we've really solved it or not. So at this point we have to solve that problem, and that's what we're aiming at.

So, the one way to be able to transmit a whole sequence of signals is to have these choices of vectors here and to develop a set of orthonormal waveforms, v_1 up to v_j , which all have the property that if you time shift them each by capital T , they're still-- if you time shift them by capital T , they have to be orthonormal to each other. The question you're facing is whether these things that you're transmitting are orthogonal to all of these things that you're transmitting. Now, in the Nyquist problem, we dealt with the problem of how do you take one waveform here and make it orthonormal to all of its time shifts. And we solved that problem. In the quiz you solved the problem-- although you probably didn't recognize it-- of dealing with orthonormal functions both in time and in frequency. And that's the kind of thing we would like to use here. If I take a set of orthonormal pulses and then I modulate those orthonormal pulses up to a higher frequency-- which is out of the range of this first frequency-- then I can send one sequence of orthonormal functions down here, another set of orthonormal functions up here in a different frequency range. Another one up here in a different frequency range and so forth.

So then all of these orthonormal functions are going to be orthonormal to each other. Yeah?

AUDIENCE: Are you saying that each x of k is its own frequency? Because each x of k is infinitely long.

PROFESSOR: Each x of k is going to-- each x of k is just a vector of j components. I'm going to modulate that vector, x of k , into a waveform, x of t , which might be finite duration or it might be infinite duration. I mean, it's going to go to zero very, very fast, anyway.

And whether it is absolutely time limited or not is something I don't really care about at this point.

But the point is I can create functions where, in fact, I have a whole sequence of functions here and they're orthonormal to all of the shifts. One way of doing this, for example, is to make capital J a bunch of little time shifts on functions. I can pick a function p of t , which is orthonormal to all of its shifts, in terms of t_1 . I can send J of those pulses to take care of x of k . And then I can use a capital T in here, which is j times this little t that I was using. And I can send another set of functions. So I can do that, I can move up and down in frequency. I can choose any old set of orthonormal functions that I want to. But the thing that I want to do is I want to make sure that for each vector, x of k , that I'm sending in time when I modulate it to a waveform, that waveform is orthonormal to the waveforms for every other k . And there are lots of ways of doing that. OK, mainly there are lots of choices of orthonormal functions.

OK so anyway what I'm going to be doing is making all of these signals orthogonal to each other. OK, so the transmitting waveform for this sequence of modulated signals is x of t , which is the sum of x of k times t minus kt . Mainly the same thing we were doing before. Except now I have also the problem that each of these waveforms, x of k of t , has to be some sum of orthonormal functions. So the problem becomes a little more difficult than what it was before. But in fact it's-- I mean this is just standard communication. Every wireless system in the world uses this kind of scheme. They don't use QAM or PAM. They use something much more like this.

OK, so now our problem is you want to detect a generic x from this sequence. OK, in other words, one of these x sub k in sequence, we want to be able to detect what signal was sent. We want to detect which hypothesis chose a signal which was then formed into a waveform, x of k of t . And if I can do this for one k , I can do it for all of them. So I want to solve the problem for one generic value of k . OK, how is this problem different from what I was looking at before? Before I was looking at the problem where we built a communication system, we tuned it all up, we sent one bit.

We detected it, we tore it all down and we went home. Now what we're doing is we're building the communication system. We're tuning it all up. We're sending a sequence of bits. And then all I'm interested in at the moment is detecting the k 'th of them. But if I find a way to detect the k 'th of them, I can then use it for every k . OK?

So I'm going to build a detector which is going to detect, in some optimal way, each one of these signals that gets sent. OK? Is it clear how the problem is different? Mainly I have to deal with the fact that these other signals are floating around there. And that's my problem.

Ok, so the input to the channel is hypothesis H . That takes values one up the m . The symbol, m , is mapped into the signal, vector $a_{\text{sub } m}$, it's modulated into x of t equals summation over j , $a_{\text{sub } m j}$, ϕ_j of t . OK, this waveform, now, is a function of which particular signal I'm sending. Which is a function of which particular hypothesis entered the encoder. The trouble with this material is all the complication comes in this awful notation, which you can't avoid. Because you're dealing with sequences, you're dealing with vectors, and you're dealing with wave forms all at the same time. What's going on, after you understand it, you'll say why was it so difficult to understand this? Because eventually when you see it, it becomes very, very simple. And I understand why there's just too much stuff all going on at the same time.

OK, so what I'm going to do now is I'm going to take these J , capital J , orthonormal waveforms. And we've already seen that you can start out with any old orthonormal waveforms and if you want to you can extend that set of waveforms into an orthonormal set that spans all of L^2 . OK? So I'm going to imagine that we've done that. It's taken us a long time, but we've done it. We're all through with it. We have this orthonormal set now.

If I'm smart, that orthonormal set, which I generated, will also include easy ways to represent each of the other signals that we're going to send. But I don't care about that right now. All I'm dealing with is this one hypothesis that came in. This one signal, $a_{\text{sub } m}$ -- oh, the hypothesis m , the signal $a_{\text{sub } m}$, at the particular time

instant, k , and this waveform that gets sent, which can be represented as the first J terms in this orthonormal sequence.

OK, so what I'm going to get then is the received random process. Is going to be a sum -- and forgot about the j prime now-- I can represent it as a sum of coefficients times these orthonormal waveforms. OK, that's what we've done for arbitrary sequences, and then we've said we can also do it for at least well defined random processes. I'm going to make, I mean, instead of making this an infinite dimensional sum, I want to make it a finite dimensional sum where J prime is very, very large. Say, 10 to the fiftieth if you want to. I don't want to make it infinite, I want to look at what happens when I let it get bigger or smaller. So I'm expanding Y of t over an orthonormal expansion. But I'm not going all the way. I'm just going to try to do maximum likelihood detection with this finite set of observations.

So I won't do quite as well as if I have all the observations, but I'll still do pretty well. We hope. OK, well so Y sub j -- the output that I see in this degree of freedom corresponding to ϕ sub j of t . Is going to be x_j -- what I sent in that degree of freedom-- plus z_j . And there are j degrees of-- capital J degrees of freedom that I'm using. So the outputs in those degrees of freedom, namely in the ϕ_1 of t , ϕ_2 of t , ϕ_3 of t directions in this L_2 space are going to be the signal plus the noise. For all of these dimensions. And Y_j is just going to be equal to z_j for all the other terms. OK, now I want to add one extra thing here. What I should be putting in here is all the other signals that are going to be transmitted. I don't know how to do that. Notationally it gets very confusing. So what I'm going to say is, OK Z sub j here is not just Gaussian noise. Z sub j is Gaussian noise plus all the signals from other time instance that we're sending. Plus all of the signals that anybody else is sending. If we're dealing with wireless then we have interference from them. Plus any old other thing you can think of. z sub j is everything but in these other degrees of freedom. In these other coordinates. This solves another problem for us, because when we defined white Gaussian noise we had this problem. That we could only say it looked white over the region of interest. We could only say it looked white over some time span. Because the earth keeps changing, you know. And over some frequency span because different frequencies behave in different ways.

So this also allows us to have different Gaussian random variables here. So when we have arbitrary random variables here, they can be Gaussian or non-Gaussian. They don't have to have the same variance. They don't have to have anything. What I am going to assume is that these out of band, out of sense, out of view random variables are all independent of the things that I am looking at. And for white Gaussian noise, that's true. All of these random variables here are independent of all of these random variables here. For these first capital J different random variables that I'm interested in.

And all these random variables are independent of they input that I'm using. OK? In other words, a hypothesis came into the transmitter that generated a signal. The signal got turned into a waveform, which is defined solely in terms of these J degrees of freedom, these J orthonormal functions. And now everything everywhere else is independent of these J functions that I'm interested in. Why is that a shaky assumption? Anybody think of a situation where that is absolute nonsense? Yeah?

AUDIENCE: The stuff from the other message had t--

PROFESSOR: Mm hmm.

AUDIENCE: You said t of j is not just Gaussian noise--

PROFESSOR: It also includes all those other signals, yes. Well, but I want to assume that those other signals are independent of this particular signal that I'm sending. But in fact that is making a pretty big assumption. Because one of the things that a lot of people like to do is, when these bits come into a channel, the first thing they do is they encode the bits for error correction. And then they take those bits that come out of the error correction device, error encoding device, which are as correlated as could be. And they're all statistically very dependent, because we want to use that statistical dependence later to correct errors.

And this assumption that I'm making here says, "no that's not the case. I'm assuming that all that other stuff is independent of what I'm transmitting here." So

I'm very specifically assuming at this point that all of that stuff has not been encoded first. That I'm sending something which is independent of everything else. Which is going to enter this channel. We'll just assume that and after we get done assuming it and seeing what the consequence of it is, we'll go back and see what it all means.

OK, so for a little more notation. I'm going to call the vector, Y , the first J , capital J , outputs. I'm going to call the vector Y prime all the other outputs. Now intuitively what we're aiming at is, we would like to say this stuff doesn't have anything to do with it. We're going to base our decision on this. But I want to prove that to you, and show you why it works and why it doesn't. And the noise I'm going to break up the same way. Z is this the first J components of the noise. And Z prime is the other components of noise. So what I have is that the output, the output that I want to look at, namely the output this vector of dimension J output. Which is equal to a vector of dimension J input plus of a vector of dimension J noise is equal to-- well Y is equal X plus Z . And the out of band stuff, the output, is just these noise and other signals.

OK? And I want to assume that Z prime, X , and Z are statistically independent. Question, test your probability. If I assume that Z prime is independent of Z , and if I assume that Z prime is independent of X , does that mean that Z prime is independent of X and Z ? If a is independent of b , and a is independent of c , is a necessarily independent of the pair bc ? How many think that's true? Better go back and study a little elementary probability again. And the notes are occasionally wrong about that, too. So you shouldn't feel, you shouldn't feel badly about it.

No, the problem is you really need this joint independence between all three of them. I could, for example, make X plus Y be equal to Z . I could do this with discrete random variables. Which are equally probably zero and one. And make the plus equal to a mod 2 operation. And if I did that, each pair would be independent of each other, and the triple would be very, very highly dependent. So anyway, I want to assume that Z prime, X , and Z are statistically independent. In other words, what I'm doing is saying, "If I assume that, what's the consequence of it?"

OK, so the likelihood then, the probability density of the output, Y -- this is the output

in the first j dimensions given a particular hypothesis Y is equal to the probability density of the noise evaluated at Y minus a_m . Where this is the signal that goes with this hypothesis, well with a_m . Times the probability density of Y prime for Z prime. OK? And I don't even have to assume that this is Gaussian. All I've done is to assume that these random variables are independent of these random variables. And therefore this probability density is multiplied by this probability density. OK, well that's kind of neat. Because if I put a different i in here in place of m , I get this thing. Changes all around. $F_{Z|Y} - a_i$. But this stuff, which is out of band, doesn't change at all. When I form the likelihood ratio, what I get then is this divided by that. What has happened to Y prime? Y prime has disappeared.

In other words, Y_1 to Y_j are a sufficient statistic for this problem. We've shown that sufficient statistics are the only thing you need to use to do maximum likelihood detection. OK? In other words, all those other signals, all that other noise, all that stuff from out of band has disappeared.

Now let's go back to the fact that we were looking at a finite dimensional problem. What happens now when I make j prime bigger? When I start enlarging j prime? What happens to all these probabilities that we're talking about? This probability density goes ape because of this term here. We're talking about a probability density here which is involving more and more and more terms. I can't talk about that. It doesn't go to any limit. It goes to absolute nonsense as j prime gets big. But if I form the likelihood ratio before I go to the limit, then I can go to the limit quite easily. Because there isn't any limit involved there. OK, in other words this is the likelihood ratio between hypothesis m and hypothesis i , if in fact I looked at this entire infinite amount of observation. This is all I need to make the optimal MAP decision.

OK, so there's a theorem here which is called the theorem of irrelevance. This is something that Wosencraft and Jacobs in their book on communication many years ago stressed a lot in trying to come up with a single space viewpoint of communication. And you'll see why this really does give you a single point viewpoint of communication. It says that assume that Z prime is statistically independent of

the pair X and Z . Then the MAP detection of X from the observation of Y and Y' depends only on Y . The observed sample value of Y' is irrelevant. OK? So you can do all of detection theory, you can do all of communication, simply forgetting about that irrelevant stuff.

Because of the theorem we can stick to finite dimensional vectors and the other signals can be viewed as part of Z' . So you don't have to worry about them. So long as each signal is independent of each other-- which means that these groups of bits, the first group a bit is used to form a sub x sub 1. The next group, the form x sub 2, the next group the form x sub 3, and so forth. So long as those sequences of bits are independent of each other, you're fine.

Now, suppose they aren't? What happens then? Interesting question. We said that if they are independent, I can really do maximum likelihood detection on the whole sequence. If they aren't independent, suppose I say, "oh I don't care about that." I'm just going to use this portion of the output to make my decision and not worry about whether this is independent of anything else. I can do that, this still is going to give me the optimal maximum likelihood detection in terms of the observation y_1 up to y sub j . So in other words, whether I have coding done before this or not doesn't make any difference. I can still use maximum likelihood detection on the basis of y_1 up to y sub j . What the theorem says is if the out of band stuff-- both these inputs and the noise-- are independent of what I'm trying to detect, maximum likelihood becomes the optimum thing to do for equally likely inputs. And otherwise, it's a perfectly reasonable thing to do but it's not optimal.

Now, a lot of people-- and we'll see some examples of this when we look at wireless-- in fact, use coding. Then they use this particular kind of detection where they forget about all of the added information from other signals. They make a decision on each of these x sub k , namely each of the M -ary signals that goes in, they make a hard decision on it. It's called a hard decision, not because it's difficult it because they refuse to ever go back and change it. If they just say likelihoods and try to put things together in the final decoder, it's called soft decoding. Otherwise it's called hard decoding. If you do soft decoding, it has to work better. Because you're

making, in a sense, a better decision because eventually you're using more information. So soft decisions are better than hard decisions. Used to be that everybody used hard decisions because hard decisions were easy and soft decisions were hard. Strange, strange thing. But anyway that's changed. Why? Well it ought to be obvious why. Because anything you build now cost a tenth of what it used to cost to build it. I heard Irwin Jacobs awhile ago saying that one of the things that they always did when they were designing new pieces of equipment is they would look at how much it would cost to build these devices. And then, as opposed to most companies which would say that's too expensive let's find the cheaper way to do it, they said OK how long is it going to take for us to do it, and what is the price of components going to be by time we got it done? And they would usually say, well it's going to cost a year before we can go into mass production. By that time, everything will cost less than a half of what it's costing now. So let's go ahead and do it the right way. So again the argument comes that you can do the hard thing now. Which is soft decisions, and that's what most people do at this point.

Let me give you one more picture to get ready for what we're doing next time. Because it's a nice picture of different signal sets. Because we've just talked abstractly of having multiple signals viewed as vectors, and this will give us some idea of what all of these mean.

I can have two signals, a binary signal set, and I can insist on the signals being orthogonal to each other. Which is a nice thing to do some times. But then I can look at it and I can say, "how can I make that a better signal system?" The trouble with this signal system is it's not antipodal. It's not antipodal because somehow by alternating between these two orthogonal signals-- there's a mean between them-- and I'm transmitting that mean plus the difference. And the difference between them is minus 0.7 and plus 0.7 in that direction that way. If you can, I guess you can't see it that way. In this direction this way. Well anyway, OK.

A better thing to do than this is this, which is called bi orthogonal. So you take orthogonal signals and then you have a signal set consisting of four signals and two dimensional space. We then talk about orthogonal signals and higher dimensional

space. You can talk about three orthogonal signals in three dimensional space here. There, there, and there. So that's m equals 3 and j equals 3.

If you do the same thing that we did here-- we're going to make this into an equilateral triangle. Namely we're going to center it around the center. If we do the same thing that we did here we're going to turn this into a set of six waveforms which are still using the three degrees of freedom, but at least get us more signals. For the same number of degrees of freedom. So you can extend this picture as far as you want to. You can talk about many, many orthogonal signals going into many more degrees of freedom. For each one of them you can come up with a simplex set of signals. The nice thing about the simplex set of signals is that all of the signals are arranged around the center point. They're all equally distant from each other. You can get these for every dimension by starting with these and simply taking the mean out, which loses you one dimension and makes this sort of ideal set. Well tomorrow-- on Wednesday what we're going to do is, we're going to talk about these large sets here, and see what happens. And we'll see that you in fact get to channel capacity this way. OK.