

**PROFESSOR:** There is one handout being passed out, it's chapters four and five, so be sure to pick it up. And just a reminder that there is a homework due on next Wednesday, so we have homeworks due every week in this course as well.

So last class, we covered Chapters One through Three quite quickly rather, because it was mostly a review of 6.450, and one of the key ideas we covered last class was the connection between continuous time and discrete time systems. So we have continuous time, discrete time. A specific example that we saw was that of an orthonormal PAM system.

The architecture for a PAM system is as follows: you have  $X_k$ , a sequence of symbols coming in, they get into a PAM modulator. What you get out is a wave form,  $X$  of  $t$ , is what the PAM modulator produces. You have a little noise on the channel, and what you receive out is  $Y$  of  $t$ .

So  $Y$  of  $t$  is the wave form that the receiver receives, and what a canonical receiver's structure to a matched filter followed by sampling. So what you get out is a sequence of symbols  $Y$  of  $t$ . So this is a structure for an orthonormal PAM modulator, and the continuous time version of the channel is that you have  $y$  of  $p$ , which is received at the receiver, plus  $X$  of  $t$ , plus  $N$  of  $t$ .

The discrete time model is you have the symbol  $Y$  of  $k$  as the output of the sampler. It equals  $X$  of  $k$  plus  $N$  of  $k$ . Now basically, we say that the two systems are equivalent in the following sense: if you want to make a detection of  $X$  of  $k$  here, at the receiver,  $Y$  of  $k$  is a sufficient statistic, given that  $Y$  of  $t$  is received at the front end of the receiver. So that's the equivalence between discrete time and continuous time.

And the way we established this fact was by using the theorem of irrelevance. The noise here is white Gaussian noise, so if we project it onto orthonormal wave forms, the corresponding noise samples would be IID. The noise will be independent of everything which is out of band, and so there is no correlation among the noise

samples, and  $Y$  of  $k$  is a sufficient statistic to detect  $X$  of  $k$ .

So that is the basic idea. Now, a similar architecture also holds when your continuous time system operates in fast band rather than base band, except the main difference now is instead of a PAM modulation, you have a QAM modulator, and your symbols,  $X$  of  $k$  and  $N$  of  $k$ , will be complex numbers rather than the real numbers.

Now the connection between continuous time and discrete time systems can be made more precise by relating some parameters in continuous time with those in discrete time. OK? So we have continuous time parameters, and discrete time parameters. So a continuous time parameter is a bandwidth,  $W$ . A discrete time parameter is given by symbol interval  $T$ , and the relation is  $T$  equals  $1$  over  $2W$ . This is for a PAM system. OK?

So something is shown in 6.450, and it's shown by using Nyquist's ISI criteria. Do not want to have ISI in the system, the maximum symbol rate, or rather the minimum symbol rate, would be  $1$  over  $2W$ . You cannot send symbols faster than this rate.

A second criteria is power, which is  $P$  here, in continuous time. In discrete time the equivalent parameter is energy per two dimensions. It's denoted by  $E_s$ , and  $E_s$  is related to  $P$  by using  $E_s$  equals  $2$  times  $PT$ . Note that this is for two dimensions, meaning in PAM we have one symbol per dimension, so this is for two symbols. This is energy for two symbols for a PAM modulator, and the relation is  $E_s$  equals  $2$  times  $PT$ .  $T$  is the time for the one symbol. We also -- we are looking at energy for two symbols, so we multiply  $P$  by  $2T$ . OK?

Noise in continuous time is AWGN, for Additive White Gaussian Noise process, and the power spectral density is flat on positive support of bandwidth, and the height is  $N_0$ .

OK, so this is how the spectral density is. In 6.450, we looked at double-sided power spectral density when the height was  $N_0$  over two, but it was going over both the

positive and negative part of the bandwidth. Here, we are only looking at the positive part of the bandwidth, and we are going to scale noise by  $N_0$ . This is just a convention.

In discrete time, your noise sequence is  $N_k$ , and is an IID and you take them as Gaussians with zero mean, and variance of  $N_0$  over 2. So we have noise which has a variance of  $N_0$  over 2 per dimension. And this was, again, shown by this theorem that when you project noise onto each of the orthonormal wave forms, you get the variance is  $N_0$  over 2. OK?

Instead of a base band system, if you had a fast band system, then instead of having a PAM modulator, we would have a QAM modulator. So if instead of PAM we had a QAM then the main difference now would be that these symbols are going to be complex numbers instead of real. So what's the symbol interval now going to be?

**AUDIENCE:**  $1$  over  $W$ .

**PROFESSOR:** It's going to be  $1$  over  $W$ .

Instead of sending a real symbol, we are sending one complex symbol, which is occupying two dimensions, and our symbol rate is going to be  $1$  over  $W$ . Well, the energy for two dimensions, is still given by  $E_s$  equals  $2PT$ , or equivalently,  $P$  over  $W$ . OK? Or that's -- The noise samples  $N_k$  are still IID, but now they are complex Gaussians, with zero mean, and variance  $N_0$ . Or equivalently, they have a variance of  $N_0$  over 2 per dimension.

So what we see here is that we can have analogous definitions for discrete time and continuous time, and one of the key parameters that comes up over and over again in the analysis is this notion of signal to noise ratio. The signal to noise ratio is defined as the energy per two dimensions, over the noise variance per two dimensions. So that's the definition of signal to noise ratio.  $E_s$ , well, it's  $2PT$ , or equivalently,  $P$  over  $W$ . The noise variance for two dimensions is  $N_0$ , so the definition is the same as -- SNR equals  $P$  over  $N_0 W$ . OK, are there any questions?

The other notion we talked about last time is this idea of the spectral efficiency. In

continuous time, the definition is quite natural. It's denoted by symbol  $\rho$ . The units are bits per second per Hertz, and it's basically  $R$  over  $W$ . You have  $R$  bits per second over  $W$  hertz. So it's the amount of information bits that I'm able to send over the amount of bandwidth that I have in my system.

In discrete time, we can also define the same idea of spectral efficiency, but it's usually -- and a good way to think about spectral efficiency from a point of view of a design of an encoder. What the encoder does, is you have an encoder here, it takes a sequence of bits in-- so say you have  $b$  bits coming in-- and it produces  $N$  symbols. So this could be  $X_1, X_2, X_n$ , and you have a sequence of  $B$  bits --  $b_1, b_2, b$  subcapital  $B$ . This is how an encoder operates. It maps a sequence of bits to a sequence of symbols.

Now where does the encoder fit into this architecture? It fits right here at the front, right? You have bits coming in, you encode them, you produce a sequence of symbols, and you send them over the channel. So if I have this encoder, what's my the spectral efficiency going to be?

Well, you have to ask what the encoder does, right? So from here, we have a PAM modulator. So it's basically this from here on, we are back to the system. So what's the spectral efficiency now for this system? How many bits do you have per symbol?

**AUDIENCE:** [UNINTELLIGIBLE]

**PROFESSOR:** You have  $B$  over  $N$  bits per symbol. Now how many symbols do you have per dimension, if this is an orthonormal PAM system?

**AUDIENCE:** One symbol per dimension.

**PROFESSOR:** You have one symbol per dimension, right? So you have  $B$  bits per  $N$  dimensions, in other words.

**AUDIENCE:** [UNINTELLIGIBLE]

**PROFESSOR:** So in QAM how many -- you have usually how many symbols per dimension?

**AUDIENCE:** [UNINTELLIGIBLE]

**PROFESSOR:** Half symbols per dimension, right. So in that case, the spectral efficiency -- the units are bits per two dimensions, is  $2B$  over  $N$ .  $B$  over  $N$  bits per dimension, because the units are bits per two dimensions, you get  $2B$  over  $N$  bits per two dimensions. OK?

Now a natural question to ask is, how is this definition related to this definition here? This is quite the natural definition, and here we have imposed this encoder structure. Are the two definitions equivalent in any way? And in order to understand this, let us take a one-second snapshot. So now in one second, I can send  $N$  equals  $2W$  symbols. Because this is an orthonormal PAM system, I can send  $2W$  symbols per second, but in one second, I can send  $N$  equals  $2W$  symbols. Because my rate is  $R$  bits per second,  $B$  equals  $R$ , in one second I can send  $R$  bits. So now my definition of  $\rho$ , which I defined to be  $2B$  over  $N$ , is same as  $2R$  over  $2W$ , which is  $R$  over  $W$ . So the definitions in continuous time and discrete time are equivalent. OK?

Now, why is spectral efficiency very important? Well, there is a very famous theorem by Shannon which gives us a nice upper bound on the spectral efficiency. Perhaps the most important theorem in communications is Shannon's Theorem, it says that if you have an AWGN system with a certain SNR, then you can immediately bound the spectral efficiency by  $\log_2$  of  $1$  plus SNR bits per two dimensions.

This is a very powerful statement. Equivalently, the capacity of an AWGN channel is  $\log_2$   $1$  plus SNR bits per second. So one important observation here is if I have a communication system, and if what I care about is the spectral efficiency of the capacity, there are only two terms that are important. One is the signal to noise ratio, which is  $P$  over  $N_0 W$ , which is defined here. So the individual units of  $P$  and  $N_0$  doesn't matter, it's only the ratio of  $P$  over  $N_0$  that matters. And the second parameter is the bandwidth  $W$  that we have in the system. So signal to noise ratio and bandwidth are in some sense fundamental to the system.

An operational meaning of this theorem is that if I look at this encoder, then it gives me an upper bound of how many bits I can put for each symbol. The number of bits that I can put on each symbol is upper bounded by this term here,  $\log_2$   $1$  plus SNR.

I cannot put arbitrarily many bits per each symbol here.

Now in order to make such a statement, there has to be some criteria that we need to satisfy. And what is the criteria for Shannon's Theorem? In order to make such a statement, I need to say that some objective function has to be satisfied. Because in some sense, I could just put any number of bits I could have on upper symbol, right? The encoder could just put 100 or 200 bits. What's going to limit me?

**AUDIENCE:** Probability of error?

**PROFESSOR:** The probability of error. So this assumes that -- OK? Now, you're not responsible for the proof of this theorem, it is in Chapter Three of the notes. Basically, it's just a random coding argument, which is quite standard in information theory. So if you already taken information theory or otherwise, you would probably have seen that argument involves bounding atypical events that happen. So the probability of error is an atypical event, and we use asymptotic equipartition property to bound the error event. There's a standard proof in Cover and Thomas, for example.

Now the main difference in Professor Forney's approach is that he uses this Theory of Large Deviations. Theory of Large Deviations basically gives you a bound on the occurrence of rare events, and it is well-known in statistical mechanics. So it's kind of a different approach to the same problem. The basic idea is same as you would find in a standard proof, but it uses -- it comes from this idea of large deviations theory.

So for those of you who are taking information theory or already have seen it, I urge you, go at some point, take a look at this proof. It's quite cool. I already saw somebody reading this last Friday, and it was quite impressive. So I urge more of you to do that.

So now that we have the spectral efficiency, a natural thing is to plot how the spectral efficiency looks like. So what I'm going to plot is as a function of SNR the spectral efficiency. Typically, when you plot SNR on the x-axis, you almost always plot it on a dB scale. But I'm going to make one exception this time, and I'm going to

plot this on a linear scale.

So this point is zero, or minus infinity dB here, and I'm going to plot -- well I should call this, actually, rho Shannon, so I don't confuse the notation. So we'll define rho Shannon as  $\log_2 1 + \text{SNR}$ . So this is rho Shannon here, and we want to plot rho Shannon as a function of SNR. Now if my SNR is really small,  $\log 1 + \text{SNR}$  is approximately linear, so I get a linear increase here. If my SNR is large, then the logarithmic behavior kicks into this expression here, so now the spectral efficiency grows slower and slower with SNR.

So this is a basic shape for my spectral efficiency. And this immediately suggests that there are two different operating regimes we have. One regime where the spectral efficiency increases linearly with SNR, another regime where the spectral efficiency increases logarithmically with SNR. So if SNR is very small, then we call this regime as power-limited regime. And if SNR is large, we call this the bandwidth-limited regime.

These are our definitions. And let's see what motivates their names. Suppose I have a 3 dB increase in my SNR, and I am in power limited regime. How does rho Shannon increase?

**AUDIENCE:** [UNINTELLIGIBLE]

**PROFESSOR:** A factor of 2, right? Basically, if I have a 3 dB increase in SNR, my SNR increases by a factor of 2. In this regime, rho Shannon increases linearly with SNR, so I have a factor of 2 increase in my spectral efficiency.

What about this regime here? If SNR increases by 3 dB, how does rho Shannon increase? It increases by one bit per two dimensions. Units of rho are bits per two dimensions. I have a logarithmic behavior kicking in, so rho is approximately  $\log \text{SNR}$ . If I increase SNR by a factor of 2, I get an additional 1 bit per two dimension scale. OK?

If I increase my bandwidth in the power-limited regime by a factor of 2. So this bandwidth here increases by a factor of 2, how does my capacity change, if I'm in

the power-limited regime? There's no change, right? Or is there a change? I'm in the power-limited regime here, and I increase my bandwidth by a factor of 2.

**AUDIENCE:** [UNINTELLIGIBLE]

**PROFESSOR:** Right. What is my SNR? It's  $P$  over  $W N_0$ . So what happens if I fix  $P$  over  $N_0$ ? OK, yeah, so you had it. Basically, if I double my bandwidth, my SNR decreases by a factor of 2, so this term here is like SNR over 2,  $W$  increases by a factor of 2, and there is no change in bandwidth.

So let's do it more slowly. So we have, say, in the -- and say, my  $C$  is now  $W \log_2 1$  plus SNR. Instead of SNR, I will write  $P$  over  $N_0 W$ . So this approximately is  $W$  times  $P$  over  $N_0 W$  times  $\log E$  to the base 2. And this is approximately -- this is basically  $P$  over  $N_0$  times  $\log E$  to the base 2. So in other words, in the power-limited regime, changing bandwidth has no effect on the capacity.

What happens in the bandwidth-limited regime? Well, in this case  $C$  equals  $W \log_2$ . Well, I can say approximately, and I can ignore this factor of 1, because  $P$  over  $N_0 W$  is large, because my SNR is large in the bandwidth-limited regime. I'm operating in this part of the graph. So this is  $P$  over  $N_0 W$ . Now suppose I increase my bandwidth by a factor of 2.  $C$  prime will be  $2W \log_2$  over  $P$  over  $2 W N_0$ . All right? This, I can write it as  $2W \log_2 P$  over  $N_0 W$  minus -- this is to the base 2, so I can write a 1 here. And this term is approximately same -- if I ignore the 1, if I assume  $P$  over  $N_0 W$  is quite large, then I can just ignore the subtraction of 1. So I get  $2W \log_2 P$  over  $N_0 W$ , or it's equivalently  $2C$ .

OK, so in other words, in the bandwidth-limited regime, if I increase my  $W$  by a factor of 2, the capacity approximately increases by a factor of 2. So that's what motivates the name bandwidth-limited regime here. Are there any questions? OK.

All right. So it turns out that there are two points I wanted to say. First of all, one might ask, that fine, these seem to be interesting definitions, that SNR is much smaller than one, and SNR is much larger than one, but is there any kind of hard division between the bandwidth-limited regime and power-limited regime? I mean,



the general answer is no. It's basically -- because there is some point when the capacity appears not strictly logarithmically, or strictly linearly in terms of SNR. But from an engineering point of view, we take  $\rho$  equals two bits per two dimension as a dividing point between the two regimes.

And one motivation to see why  $\rho$  equals two bits per two dimension is a good choice is because this is the maximum spectral efficiency we can get from binary modulation. OK, if you want to have spectral efficiency more than two bits per two dimensions, you have to go to multi-level modulation, and that's one of the reasons why this choice is often used in practice.

Another point is that the bandwidth-limited regime and power-limited regime, they behave quite differently in almost all criterias that you can think about. So usually, with the analysis -- we keep them separately and do the analysis differently in the bandwidth and power-limited regime, and that's what we will be doing in the subsequent part of this lecture and next lecture.

So we start with the power-limited regime. Now, we already saw that in doubling the power doubles  $\rho$ , and doubling bandwidth does not change  $C$ .

OK, the other point I mentioned was binary modulation is sufficient in this regime, because our spectral efficiency is less than two bits per two dimensions, so the idea is to have a strong code followed by binary modulation. What else?

Right. Typically, the normalization is done in per information bit. What does this mean? When we are wanting to compare different systems, we will look at all the parameters normalized per information bit. For example, if we want to look at the probability of error, we look at this quantity here,  $P_b$  of  $E$ , which is the probability of error per information bit, as a function of  $E_b/N_0$ . This is an important trade-off that we wish to study in power-limited regime.  $E_b$  is the energy per bit,  $P_b$  of  $E$  is the probability of error per information bit.

So I want to spend some time on this  $E_b/N_0$ , because it's an important concept that we'll be using often in the course. So what is  $E_b/N_0$ ? It's sometimes also

known as  $E_b/N_0$ , so it depends how you wish to call it. What is energy per bit in terms of  $E_s$ ?

**AUDIENCE:** [UNINTELLIGIBLE]

**PROFESSOR:** Well,  $E_s$  over  $\rho$ , right? Energy per bit. Well,  $E_s$  is energy per two dimensions.  $\rho$  is bits per two dimensions, so  $E_b$  is  $E_s$  over  $\rho$ , and you have  $N_0$  here. And  $E_s$  over  $N_0$  is also our SNR, so this is SNR over  $\rho$ . OK, so that's how  $E_b/N_0$  is defined.

Now we know from Shannon that  $\rho$  is always less than  $\log_2 1 + \text{SNR}$ , or equivalently,  $2^{\rho} - 1$  is less than SNR. If I sub in here, this means that SNR is greater than  $2^{\rho} - 1$  over  $\rho$ , and then we're just dividing by  $\rho$ . So  $E_b/N_0$  is always greater than  $2^{\rho} - 1$  over  $\rho$ .

And this is quite an interesting observation. For example, if you are analyzing the feasibility of a communication system, which has a certain spectral efficiency, and a certain  $E_b/N_0$ , then you can immediately check this relation, to see if it is a system in the first place. What Shannon says is that  $E_b/N_0$  is always going to be greater than  $2^{\rho} - 1$  over  $\rho$ . So if you see that this relation is not satisfied, immediately know something is wrong.

This actually reminds me of an interesting anecdote that Professor Forney once mentioned when I was taking the class. Well he has been in this field since the '60's, and so he has seen a lot of this stuff. He was saying when turbo codes -- which is one of the first capacity approaching codes in recent years when they were proposed -- they presented the results at ICC, the International Conference in Communications, and what they saw was that the performance was very close to the limit that we can predict.  $E_b/N_0$  was very close to the ultimate limit -- the  $E_b/N_0$  achieved by turbo codes was very close to the ultimate limit that one can predict, at least 3 dB better than the best codes that were available there.

So most people just thought that there was something wrong in the simulations, and they told them that there was a factor of 2 missing somewhere, so they should just

double check. But it turned out when they went back and people actually implemented these codes, they were actually very close to the capacity. So sometimes, you have to be careful. If you are not going below this limit, it could be that the system is good. And when it is, it's a really important breakthrough. Ok

So one particular concept that comes here is the idea of ultimate Shannon limit. And basically, if we are talking about power limited regime, our SNR is very small. So our spectral efficiency is going to be quite small. Now notice that this function here it monotonically increasing in rho. It's easy to show that this function here increases monotonically in rho. You could just differentiate this, or there's easier ways to do a Taylor expansion of  $2^{\rho}$  minus 1, the series expansion and show each term is positive.

So if this term is always going to be greater than this term here. So I'm going from here to here. Limit rho tends to zero of  $2^{\rho}$  minus 1 over rho. This is going to be -- it's a simple calculus exercise to show this is the natural log of 2, or in dB, it's minus 1.59 dB.

So no matter what system you design, your  $E_b/N_0$  is always going to be greater than minus 1.59 dB. That's basically what this calculation shows. And when is it achieved? Well, it's only achieved when the spectral efficiency goes to zero. So if you have a deep space communication system, where you have lots and lots of bandwidth, and you do not care about spectral efficiency, then if your only criteria is to minimize  $E_b/N_0$ , then you can design your system accordingly, check how much  $E_b/N_0$  you require for a certain probability of error, and see how far you are from the ultimate Shannon limit.

So in this way, in the power-limited regime, you can quantify your gap to capacity, if you will, through this ultimate Shannon limit. OK, are there any questions?

OK. So now let's look at the bandwidth-limited regime. So we already saw two things in the bandwidth-limited regime. If I double P, my spectral efficiency increases by one bit per two dimension. Similarly, if I double bandwidth by C, capacity approximately doubles.

Now, because you want a spectral efficiency of more than two bits per two dimension, in this system you typically do a multi-level modulation. So for those of you who are familiar, we do things like trellis-coded modulation, and bit-interleaved coded modulation, and so on. If we have time, we'll be seeing those things towards the very end of the course. This is not a subject of the course as such.

The normalization in this regime is done for two dimensions. So if you want to normalize all the quantities, we normalize them for two dimensions here. And particularly, we will be seeing probability of error as a function of this quantity called SNR norm. So this is the performance analysis that is done in bandwidth-limited regimes. Here  $P_s$  of  $E$  is the probability of error for two dimensions.

What is SNR norm? It's defined to be SNR over  $2$  to the rho minus  $1$ . Why do we divide by  $2$  to the rho minus  $1$ ? Well, that's the minimum SNR that you require for the best possible system. That's what Shannon says, right? So this quantity here is always going to be greater than  $1$ , or  $0$  dB.

So OK, well, this is the ultimate Shannon limit in the bandwidth-limited regime. If you have a system that is designed, that operates at a certain SNR and a certain spectral efficiency, you can calculate SNR norm and see how far you are from  $0$  dB. If you're very close to  $0$  dB, then that's great. You have a very good system in practice. If not, then you have room for improvement. And so, in other words, SNR norm defines the gap to capacity.

OK, so let's do an example. Suppose we have an M-PAM system. So we have an M-PAM constellation. So how does it look like? Well, you have points here on a linear line. This point is say minus alpha, alpha, three alpha minus three alpha, all the way up to  $m$  minus one alpha, and here, minus of  $m$  minus one alpha. Assume  $m$  is an even number. So the distance between two points here is two alpha, any two points.

Now we want to find the SNR norm given that we are using this constellation. So in other words, if I use this constellation in my communication system, how far am I

operating from the ultimate Shannon limit? OK, that's the question.

So in order to find what we need to find, is first energy for two dimensions. Does anybody remember the formula for M-PAM? Well, there was a very nice way of doing it in 450. One natural way is to simply sum up all of the coordinates here, square them, and divide by M. And because it's for two dimensions, we could do it -- we're to multiply by a factor of two, because it's for two dimensions-- and summation of  $X_k$  squared. And if you work out, you will get some answer here.

Another way that was shown in 450 --

**AUDIENCE:** [UNINTELLIGIBLE]

**PROFESSOR:** With uniform quantization, exactly. So the idea here is, you have a source, say, which is uniformly distributed between  $M\alpha$  and  $-M\alpha$ . So you have a source uniformly distributed, and it can be easily seen by inspection that M-PAM is the best quantizer for this particular source. Ok. So the interval regions are just of equal width. Each interval region is width  $2\alpha$ , so your mean square error, if you will, is  $2\alpha^2/12$ , so it's  $\alpha^2/3$ . Well, the variance in your source, which I will denote by  $\sigma^2$ , is  $2M^2\alpha^2/12$ , or it's  $M^2\alpha^2/3$ .

So your energy per symbol -- it's energy in your quantizer, so I'm denoting it by  $E_A$  in order to differentiate it by  $E_s$ , because  $E_s$  is two times  $E_A$ , OK? It's going to be  $m^2 - 1$  times  $\alpha^2/3$ . So  $E_s$  is  $2E_A$ , and so we get it's  $2\alpha^2(m^2 - 1)/3$ .

OK, so can anybody tell me what the spectral efficiency will be for the system, if I use an M-PAM constellation? Well, how many bits per symbol do I have?

**AUDIENCE:**  $\log_2 M$ .

**PROFESSOR:**  $\log_2 M$  bits per symbol. So since it's bits per two dimensions, the sum would be  $2 \log_2 M$  to the base 2. Ok? So right now, we have pretty much everything we need to find SNR norm. The SNR here is  $E_s/N_0$ , so it's  $2\alpha^2(m^2 - 1)/3$ ,  $M^2\alpha^2$

minus 1, over  $3N_0$ . But remember,  $N_0$  by definition is  $2\sigma^2$ , because  $\sigma^2$  is  $N_0$  over 2. That's how -- that's the noise variance per dimension. So, I had 3 times 2 times  $\sigma^2$ , or I will just write this as 6 times  $\sigma^2$ , and that's going to be -- this is just multiplication, so I should not even write it -- six  $\sigma^2$ . So this is  $\alpha^2$ ,  $M^2$  minus 1 over 3  $\sigma^2$ .

OK, so now SNR norm is  $\text{SNR}$  over  $2$  to the  $\rho$  minus 1.  $2$  to the  $\rho$  minus 1 is just  $M^2$  minus 1. It cancels with this  $M^2$  minus 1, and so I get  $\alpha^2$  over 3  $\sigma^2$ . So that's the SNR norm if I use an M-PAM system.

**AUDIENCE:** Why is [INAUDIBLE]

**PROFESSOR:** Well,  $N_0$  -- I've plugged in for SNR 3 here, so I have  $3N_0$ , but  $N_0$  is  $2\sigma^2$ . Did I miss anything? Ok? Any questions?

OK, so there are two important remarks from this example. The first remark is that SNR norm is independent of  $M$ . So I started with an M-PAM constellation, so it's a different constellation for each value of  $M$ , right? If I look at my spectral efficiency  $\rho$ , it's different for each value of  $M$ , because I can pack more and more bits per symbol as I increase  $M$ . If I look at my signal to noise ratio, it's also a function of  $M$ . But when I took SNR norm, remarkably, the  $M^2$  minus 1 term cancelled out in the numerator and denominator, and what I was left with was something independent of  $M$ .

So this is actually quite an interesting result. What it says is suppose I design a system, an M-PAM system, that has this particular spectral efficiency, then my gap to capacity is given by this expression. If I use a different value of  $M$ , my gap to the ultimate Shannon limit is still given by this expression here. So by increasing value of  $M$  or decreasing the value of  $M$ , my gap to the Shannon limit is still going to be the same. For each value of  $M$ , I will have a different spectral efficiency, but I'm not getting any kind of coding gain, if you will, here.

OK, so this motivates that M-PAM is an uncoded system. All of them have the same

gap to the Shannon limit, regardless of what the value of  $M$  is, and so.

The second point to note is if I look at the value of  $\alpha^2$  over  $3\sigma^2$ , I can make it quite small, right? I can even, if I decrease  $\alpha$  really by a great amount, I can make this quantity smaller than 1. OK? I told you here that SNR norm is always greater than 1. So what happened? Did I lie to you here, or did I do something wrong here? I mean, I can choose any  $\alpha$ , right, and make this quantity as small as I please, and then I'm doing better than the Shannon limit.

**AUDIENCE:** [UNINTELLIGIBLE]

**PROFESSOR:** Right. So basically, what's missing in this calculation is probability of error. If I make  $\alpha$  really small, what I have is all these points come closer and closer, and sure, I seem like I'm doing very well at the encoder. But what happens at the decoder? There is noise in the system, and so I get too many errors. This lower bound clearly assumes that you can make your probability of error quite small, arbitrarily small, right? So in any reasonable system, I should also look at the probability of error. If I make  $\alpha$  really small, I'm going to have too much error at the decoder, and so I won't be able to have a practical system.

So the comment is SNR norm cannot be seen in isolation. We need to couple SNR norm with the corresponding probability of error. Yes?

**AUDIENCE:** [UNINTELLIGIBLE]

**PROFESSOR:** What I mean by uncoded syst -- that's a good question. What do I mean by uncoded system? All I'm really saying here is M-PAM system is a fairly simple system to implement, right? Irregardless of what value of  $M$  I get, I have a certain gap to the Shannon limit, which is independent of  $M$ . So if I start with a simple system, where I have bits coming in, each bit gets mapped to one symbol, and I send it over the channel, I have a fixed gap to the Shannon capacity. And this, I will call an uncoded system. My only hope will be to improve upon this system. OK? Any other questions?

**AUDIENCE:** So, if I'm multiplying [INAUDIBLE]

**PROFESSOR:** Right, right. I'm assuming the fixed alpha.

**AUDIENCE:** [UNINTELLIGIBLE]

**PROFESSOR:** I mean, basically, you will see that alpha is a function of  $E_s$ , right? And I want to -- so why do I want -- the question is, why do I want to keep alpha fixed, right?

**AUDIENCE:** Yeah.

**PROFESSOR:** OK, so in order to understand that, I have to look at energy for two dimensions. If I normalize by  $M$  squared minus one -- that doesn't work out.

**AUDIENCE:** [UNINTELLIGIBLE] so the constellation cannot be expanding [UNINTELLIGIBLE]

**PROFESSOR:** Right, so I wanted to always plot by keeping alpha fixed. And I mean, the point is, typically you want to plot the probability of symbol error -- do I have that expression anywhere? Right there.  $P_s$  of  $p$  is a function of SNR norm, right? And that will be -- if I increase SNR norm, so it will be some kind of a graph, right? A trade-off. Basically, alpha will define the trade-off between SNR norm and  $P_s$  of  $p$ . So as I slip over larger and larger values of alpha, for different values of  $M$ , then I will have a trade-off as  $P_s$  of  $p$  is a function of SNR norm. Does that make sense?

So say I fixed a value of  $M$ , OK? I define SNR norm, which is  $\alpha^2$  over  $3\sigma^2$ . For this, I will get a certain probability of error now. So if I fix this value of SNR norm, I get a certain probability of error. What if I want to increase my SNR norm? The only way to do that will be to increase alpha. Does that make sense? Yeah.

**AUDIENCE:** [UNINTELLIGIBLE] You're defining SNR norm as the gap to capacity, but it seems like, I mean, obviously as you increase alpha, as you increase your signal energy, you're going to do better and better, right?

**PROFESSOR:** Right.

**AUDIENCE:** Well, in terms of what? In terms of probability of error, like achievable probability of



error, or?

**PROFESSOR:** Right, exactly. So basically, the point is say I plot  $P_s$  of  $E$  as a function of SNR norm, or as a function of  $\alpha^2$  over  $3\sigma^2$ . So my curve will look like this.

**AUDIENCE:** But the gap to capacity is defined all the way over here on the left, is that what--?

**PROFESSOR:** Right, that's a very good question. You are going much ahead than what I thought. So basically, at zero, SNR norm is zero, right? This is in linear scale, so when I have one, SNR norm is one. I have the Shannon system. Basically, for any SNR norm greater than 1, what Shannon says is your probability of error will be arbitrarily small, and here, it will be large. So this here is the Shannon limit.

And now, in a practical system, what I want to do is I want to fix the probability of error. So say I like probability of error of ten to the minus five. So that's something that the system specifies. And from that, I know the gap to the capacity. So if here, if I require, this is my SNR norm, then this is going to be my gap. I wanted to cover it, but later. That's a good point.

**AUDIENCE:** [UNINTELLIGIBLE]

**PROFESSOR:** Right, this is a certain, specific system, the M-PAM system. What Shannon says that, if you're anywhere here, you should be able to make your probability of error arbitrarily small. So your curve should basically look something like this, if you will, or even steeper.

**AUDIENCE:** What's the difference between that curve and the wider curve?

**PROFESSOR:** This curve and this curve?

**AUDIENCE:** Yeah.

**PROFESSOR:** This is basically what we achieved by an M-PAM system. I don't want to go into too much of this, because we'll be doing this later. This is the next topic. Ideally, what we would want is something, basically, this is my Shannon limit, I want something

such as the probability of error decreases right away as soon as I am just away from the Shannon limit. All right.

**AUDIENCE:** [UNINTELLIGIBLE]

**PROFESSOR:** Right. So that's a good point. So if I did not do coding, this is what I get. If I did coding, this is what I would get. So this is how much I can expect a gain from coding.

**AUDIENCE:** So in other words, basically, this bound is basically in my channel, the SNR is such that -- I mean, I'm so energy limited that my SNR norm is lower than a certain amount, there's nothing I can do, basically, is what it's saying.

**PROFESSOR:** Right. Because if you want a certain spectral efficiency, there's also spectral efficiency, right? And if your SNR is much lower, you're out of luck.

**AUDIENCE:** So basically, you have to make your spectral efficiency higher -- or, lower.

**PROFESSOR:** Lower, right. Exactly.

So now let us start with the probability of error analysis, and we'll start in limited regime. So we have  $E_b$  of  $E$  as a function of  $E_b/N_0$ , we want to quantify this trade-off. Basically, what we are doing now is trying to quantify how this graph will look like, at least for the uncoded systems today. We'll do coding next class, or two classes from now, as we get time.

So say I have a constellation,  $A$ , as binary PAM. So it takes only two values, minus  $\alpha$  and  $\alpha$ . So I have these two points here,  $\alpha$  and minus  $\alpha$ . My receiveds are  $Y$ , symbol  $Y$  is  $X$  plus  $n$ , where  $X$  belongs to  $A$ , and  $N$  is Gaussian zero mean, variance  $\sigma^2$ , where  $\sigma^2$  is  $N_0$  over 2. And now what want to do at the receiver is given  $Y$ , you want to decide whether  $X$  was  $\alpha$  or minus  $\alpha$ .

That's a standard detection problem. So how will the probability of error look like? So suppose I transmit  $X$  equals  $\alpha$ , my conditional density of  $Y$  -- let me make some room here -- will be a bell shaped curve, because of the Gaussian noise. So

this is  $P$  of  $Y$  given  $X$  equals  $\alpha$ . Similarly, if I have set  $X$  equals minus  $\alpha$ , what I get is something like this. This is  $p$  of  $Y$  given  $X$  equals minus  $\alpha$ . Excuse my handwriting there.

And the decision region is at the midpoint,  $Y$  equals zero. This is a standard binary detection problem. If  $Y$  is positive, you will say  $X$  equals  $\alpha$  was sent. If  $Y$  is negative, you will say  $X$  equals minus  $\alpha$  was sent. And your probability of error is simply the probability of error under this -- graph under these two curves here.

So we want to find what the probability of error is. Just let me just do quickly the calculation, in order to remind you. We'll be doing this over and over again. We'll just do it once. Without loss in generality, I can say this is probability of error, given  $X$  equals minus  $\alpha$ , where both the points are equally likely. So by symmetry, I can say that. So this is the same as probability that  $Y$  is greater than zero, given  $X$  is minus  $\alpha$ . Now  $Y$  is  $X$  plus  $N$ , so this is same as probability that  $N$  -- yeah capital  $N$  -- is greater than  $\alpha$ . Since  $N$  has zero mean variance  $\sigma^2$ , this is a standard  $Q$  function of  $\alpha$  over  $\sigma$ . OK, so that's the probability of error.

Now, there is one bit per symbol for each  $X$  that is one bit, so probability of error is also same as probability of bit error, so this is also  $P_b$  of  $E$ . So I have  $P_b$  of  $E$  equals  $Q$  of  $\alpha$  over  $\sigma$ , and now, I want to basically express it as a function of  $E_b/N_0$ , right. So what is  $E_b$  for the system? It's going to be  $\alpha^2$ .  $\sigma^2$  is now  $N_0$  over 2, so  $\alpha^2$  over  $\sigma^2$  is  $2 E_b$  over  $N_0$ . So now my probability of bit error is this  $Q$  function of  $\sqrt{2 E_b}$  over  $N_0$ .

OK? So let us plot this. So I'm on the x-axis, I'm plotting  $E_b/N_0$ , which is in dB scale. On the y-axis, I'm going to plot the probability of bit error. Typically, what you do is you plot the y-axis on a semi-log scale. So this is ten to the minus six, ten to the minus five. If you want to do this in MATLAB, you can use this command `semilog y`, and that does it. And so on.

So can anybody say what will be a good candidate for the x-value at this point here?  
So what should be the  $E_b/N_0$  at this point?

**AUDIENCE:** [UNINTELLIGIBLE]

**PROFESSOR:** It should be the Shannon limit, right? You cannot hope to go below the Shannon limit. So now, what Shannon says is that -- so I'm going to plot this as my Shannon limit here. This probability of error will basically use the standard waterfall curve. This is 2-PAM. And if you look at the x-coordinates, then the value at ten to the minus five is 9.6 dB.

So as a system designer, if you care about probability of error at ten to the negative five, then your gap to capacity, or gap to the ultimate limit, if you will, will be this, 9.6 minus of minus 1.59 dB. OK, I should not erase this.

OK, so the first thing is at N to the minus five, our gap to the ultimate limit is 9.6 plus 1.59 dB, and that's approximately 11.2 dB. OK?

But there is one catch to this. This particular system has a spectral efficiency of two bits for two dimensions, whereas if you want to achieve something close to the Shannon limit, you have to drive the spectral efficiency down to zero. So you might say that this is not a fair comparison. So if you do want to make a fair comparison, you want to fix rho to be two bits for two dimensions. So if you fix rho for two bits per two dimensions, you will get a limit somewhere, not at 1.59 dB, but at some other point. This is if you fix rho<sup>2</sup>, two bits for two dimensions. Can anybody say what that point will be?

**AUDIENCE:** [UNINTELLIGIBLE]

**PROFESSOR:**  $\frac{3}{2}$ . 1.5. What will it be in the log scale?

**AUDIENCE:** [UNINTELLIGIBLE]

**PROFESSOR:** 1.76. Good. So what we do know -- if we do the calculation here. If you fix rho<sup>2</sup>, two bits for two dimensions, your  $E_b/N_0$ , we know, is greater than  $2^{\frac{1}{\rho}}$ , which is  $\frac{3}{2}$ . And that is, if you remember your dB tables, 1.76 dB. Log of 3 is like 4.7, log of 2 is like 3, so it's 1.76 dB.

So in this case, your gap to the ultimate limit is going to be 9.6 minus 1.76, which

comes out to be 7.8 dB. OK?

So now, let us do the bandwidth-limited regime. Now in bandwidth-limited regime, the trade-off is  $P_s$  of  $E$  as a function of SNR norm. OK, so that's the trade-off we seek for. And the baseline system we will be using is an M-PAM system. So now, your constellation is going to be  $-\alpha$ ,  $\alpha$ ,  $3\alpha$ ,  $5\alpha$ , and so on, up to  $(M-1)\alpha$  -- I always run out of room here --  $-(M-1)\alpha$ . This is your constellation.

Now what's the probability of error going to be for this constellation? Well, what's the probability of error for each intermediate point? There are two ways you can make an error. Say we say  $\alpha$ , your noise is either too small, so it takes you to  $-\alpha$ . Your noise is too high, so it takes you to  $3\alpha$ . For each one, the probability will be  $Q$  of -- this distance is  $2\alpha$ , it would be  $Q$  of  $\alpha/\sigma$ , because you will be having your decision regions right here.

So in other words, for each intermediate point -- and there are  $M-2$  of this -- your probability of error would be  $2$  times  $Q$  of  $\alpha/\sigma$ . OK? And how many of them there are? There are  $M-2$  of these. And if all the points are equally likely, and you want to find the probability of error, you divide by  $M$ .

For the two end points, the noise can only make an error in one direction. So you get  $2/(M-1)$  times  $Q$  of  $\alpha/\sigma$ . You work this out, and you have  $2(M-1)/M$  times  $Q$  of  $\alpha/\sigma$ .

Now, we want to find probability of error for two symbols, because that's what  $P_s$  of  $E$  is, OK? So what's  $P_s$  of  $E$  going to be? Well, it's -- in terms of  $P_r$  of  $E$ , it's going to be  $1 - (1 - P_r)^2$ . This is the probability you make error in none of the two symbols, and  $1 - (1 - P_r)^2$  will be the probability of error you make in at least one symbol. And that's going to be equal to  $2P_r$  of  $E$ , or it's  $4(M-1)/M$  times  $Q$  of  $\alpha/\sigma$ .

Good, I did not write on this board. So now what remains to do is to relate  $\alpha/\sigma$  to SNR norm. And I had it on this board here. SNR norm, we just did in

the previous example, for an M-PAM system is  $\alpha^2 / (3\sigma^2)$ . OK? So I plug that in here, so I get  $P_s$  of E is  $4(M-1)/M$  times Q of root 3 SNR norm. And if M is large, this is approximately 4 times Q of root 3 SNR norm.

So we are plotting now probability of error as a function of SNR norm, similar to what we did in that part in the power-limited regime there. So this is  $P_s$  of E as a function of SNR norm. Now my Shannon limit would be at 0 dB, so this is my Shannon limit point. This is ten to the minus six, ten to the minus five, ten to the minus four, ten to the minus three, ten to the minus two, and so on. This is going to be the performance of M-PAM.

And again, I turn to the minus five, which will be the kind of performance criteria we'll be using throughout this course. You'll see that this y-axis is 8.4 dB. So in this case, the gap to capacity is going to be 8.4 dB. So if I want a certain spectral efficiency and I use M-PAM, I'm 8.4 dB away from the Shannon limit.

The idea behind coding, as someone pointed out, was to start from here and do coding to come closer and closer to bridge this gap. OK? So are there any questions? We are almost end time. OK. I think -- yes?

**AUDIENCE:** A perfect code would just be literally like a step function --

**PROFESSOR:** Right.

**AUDIENCE:** -- all the way down.

**PROFESSOR:** That's what Shannon coded. OK, I think this is a natural point to stop. We'll continue next class.