Unit 6: Noise and Symbol Demodulation

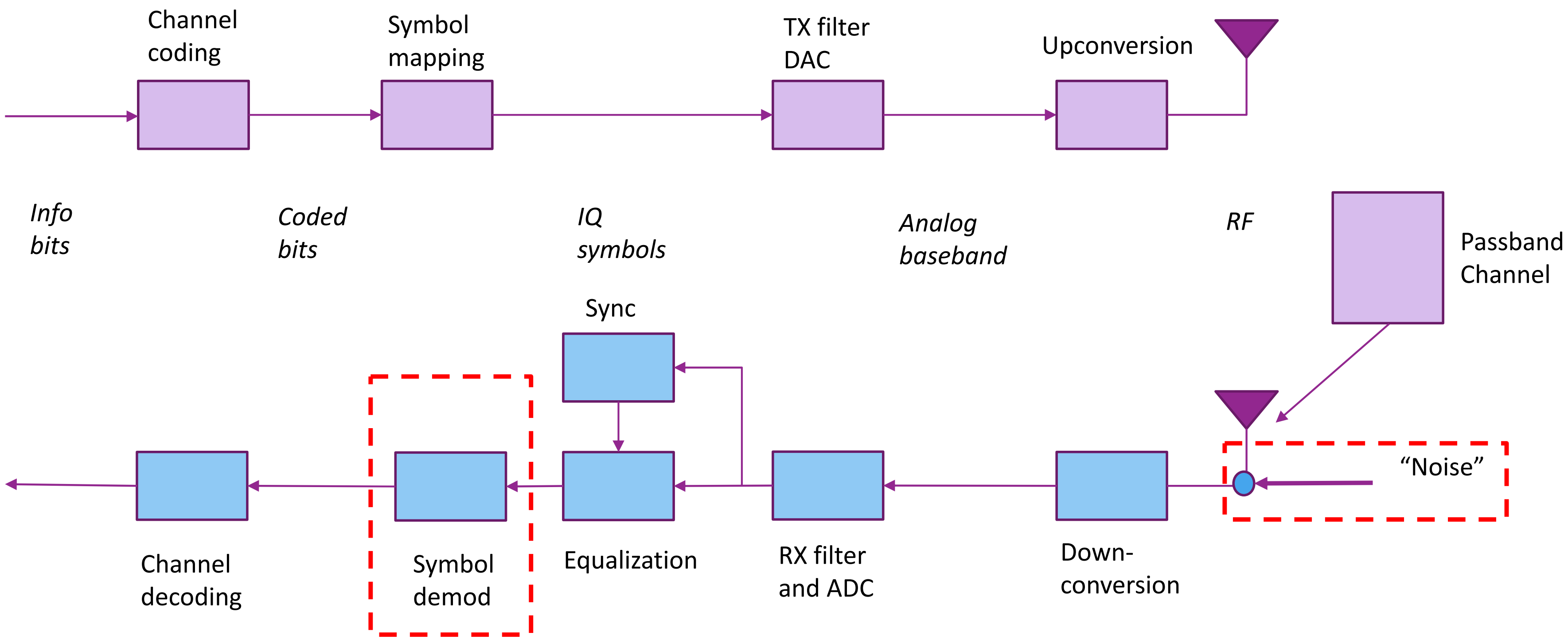
EL-GY 6013: DIGITAL COMMUNICATIONS

PROF. SUNDEEP RANGAN

Learning Objectives

- Mathematical describe AWGN noise
- Compute AWGN noise levels at passband, baseband and sample domain
- Write the ML detector given likelihoods, compute error probabilities
- Compute the ML detector for symbol detection
- Compute BER and SER probabilities

This Unit



Channel
coding
Symbol
mapping
TX filter
DAC
Upconversion
Info
bits
Coded
bits
IQ
symbols
Analog
baseband
RF
Passband
Channel
Sync
"Noise"
Channel
decoding
Symbol
demod
Equalization
RX filter
and ADC
Down-
conversion

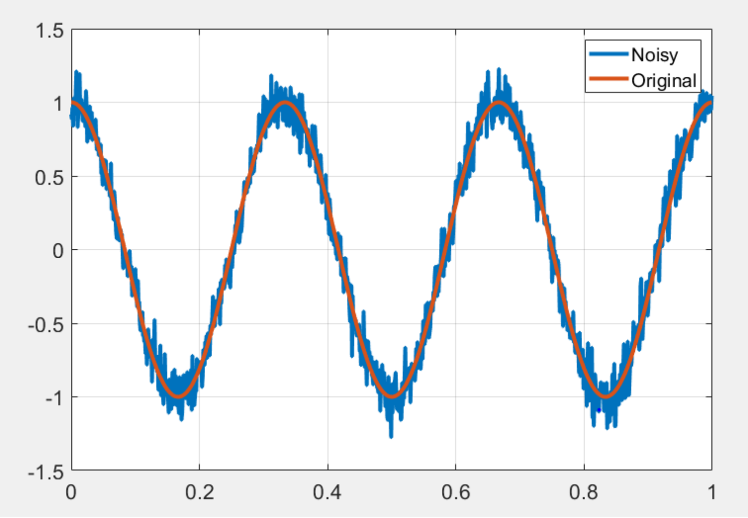
Outline

- ❑Passband and baseband noise, signal to noise ratio
- ❑Noise in the discrete symbols
- ❑ML Detection
- ❑Symbol detection
- ❑Probability of error

What is Noise?

- Noise: Any unwanted component of the signal

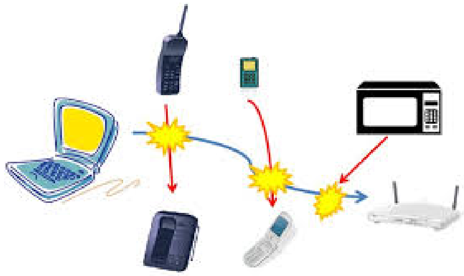
- Key challenge in communication:
  - Estimate the transmitted signal in the presence of noise



Types of “Noise”

- Internal / thermal noise:
  - From imperfections in the receiver
  - Thermal noise: From random fluctuations of electrons
  - Other imperfections: Phase noise, quantization, channel estimation errors

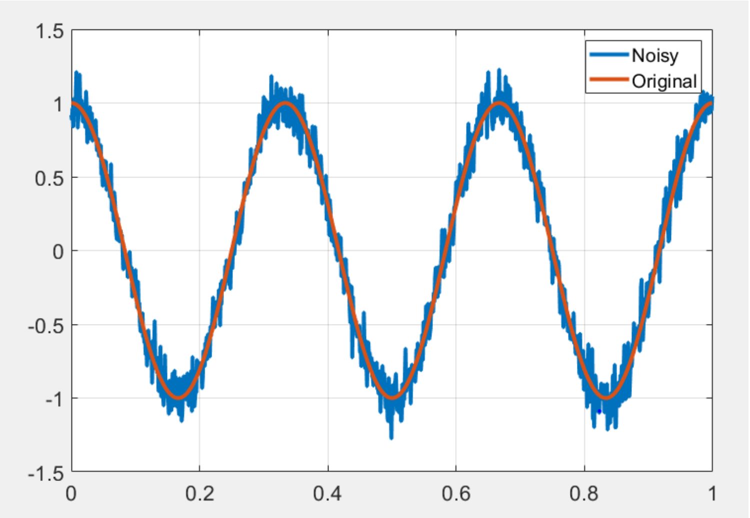
- External Interference
  - Signals from other sources
  - In-band: Transmitters in the same frequency
    Ex: Multiple devices in a cellular band
  - Out-of-band: From leakage out of carrier
  - Some texts do not consider “interference” as noise



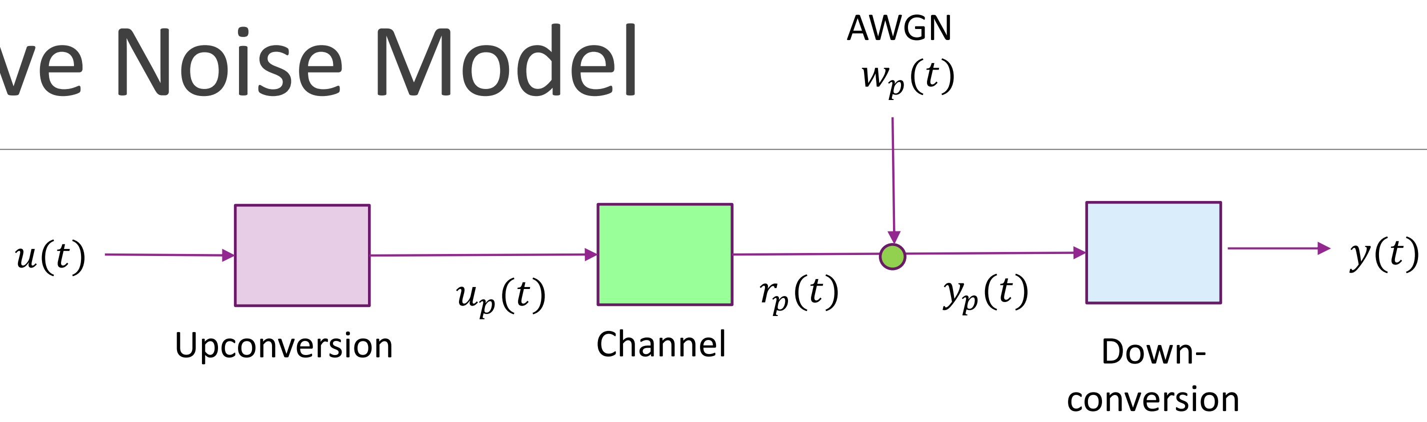
Statistical Models for Noise

- In communications, we model noise as a random process
  - Captures "uncertainty" in the value

- This lecture:
  - Describe mathematical models for noise
  - Describe effect of noise on



Additive Noise Model



- We first look at modeling thermal noise
- Thermal noise:
  - Due to random fluctuations of electrons in the receiver
  - Called "thermal" since the level of the fluctuations increases with temperature
- Common Additive White Gaussian Noise (AWGN) model: $y_p(t) = r_p(t) + w_p(t)$
  - $w_p(t)$ is real Gaussian WSS noise with PSD $\frac{N_0}{2}$

Thermal Noise

- Thermal noise: Caused by random fluctuations of electrons
- Fundamental limit determined by statistical physics: $N_0 = kT$
  - k = Boltzman constant, T = temperature in Kelvin
  - At room temperature (T=300 K), $10\log_{10}(kT) = -174$ dBm/Hz
- Practical systems see higher noise power due to receiver imperfections

$$N_0 = 10\log_{10}(kT) + NF \quad (\mathrm{dBm/Hz})$$

  - NF = Noise figure
  - Typical values are 2 to 9 dB in most wireless systems
- More in a wireless class

Scaling Up- and Down-Conversion

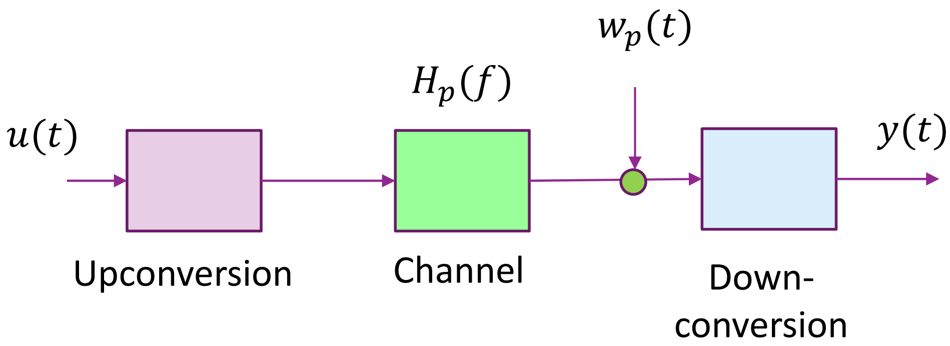
- For noise modeling, it is convenient to use a different scaling convention
- Modified scaling will keep powers in passband and baseband equal
- Note: Proakis uses original scaling and has a factor of 2 in the conversion

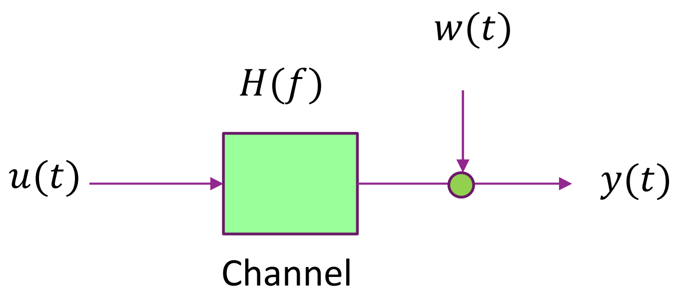
	Earlier scaling	Current scaling
Upconversion	$u_p(t) = Real(u(t)e^{j\omega_c t})$	$u_p(t) = \sqrt{2}Real(u(t)e^{j\omega_c t})$
Downconversion	$v(t) = 2u(t)e^{-j\omega_c t}$ $u(t) = h_{LPF}(t) * v(t)$	$v(t) = \sqrt{2}u(t)e^{-j\omega_c t}$ $u(t) = h_{LPF}(t) * v(t)$

Downconverting Noise

- Suppose that $w_p(t)$ is real-valued WSS noise with PSD $\frac{N_0}{2}$
- Consider downconversion (with modified scaling factor):
  - $v(t) = \sqrt{2}e^{-j\omega_c t} w_p(t)$
  - $y(t) = h_{LPF}(t) * v(t)$
- Theorem: PSD of $y(t)$ is $S_y(t) = N_0 |H_{LPF}(f)|^2$
- Why?
  - $E\left(v(t)v^*(s)\right) = 2e^{-j\omega_c(t-s)} E\left(w_p(t)w_p(s)\right) = 2e^{-j\omega_c(t-s)}\delta(t-s)\frac{N_0}{2} = N_0\delta(t-s)$
  - So $v(t)$ is complex white WSS with PSD N_0. $S_v(f) = N_0$
  - $S_y(f) = |H_{LPF}(f)|^2\, S_v(f) = |H_{LPF}(f)|^2 N_0$

Equivalent Channel with Noise





- Passband model:
  - $y_p(t) = h_p(t) * u_p(t) + w_p(t)$
  - $w_p(t)$: additive noise in passband
  - Noise PSD = $\frac{N_0}{2}$

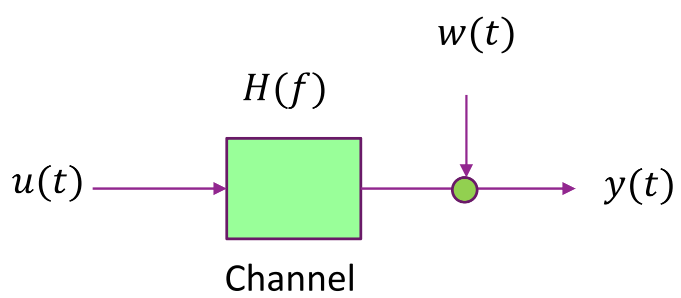
- Complex baseband equivalent model:
  - $y(t) = h(t) * u(t) + w(t)$
  - PSD of effective baseband noise:
    $$S_w(t) = N_0 |H_{LPF}(f)|^2$$

Effective Baseband Noise ≈ White

- Prev. slide: PSD of effective baseband noise is:

$$S_w(f) = N_0 |H_{LPF}(f)|^2$$

- Suppose that $|H_{LPF}(f)| \approx 1$ for $|f| \leq \frac{W}{2}$
  - Approximately constant in band of interest
- Hence: $S_w(f) \approx N_0$
- Effective baseband PSD is approximately flat
- Can be well modeled as additive white noise



Thermal Noise and Bandwidth

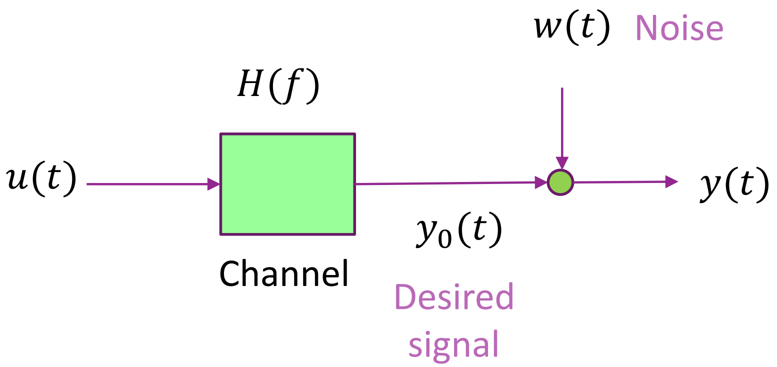
- Let $w(t)$ be the down-converted, filtered noise
- PSD $S_w(f) = |H_{LPF}(f)|^2 N_0$
- If $|H_{LPF}(f)|^2$ is an ideal LPF with bandwidth W, total noise power is:

$$P_w = \int_{-\infty}^{\infty} |H_{LPF}(f)|^2 N_0 df = \int_{-W/2}^{W/2} N_0 \, df = N_0 W = kTW(NF)$$

  - Power = Noise PSD x Bandwidth
- Example:
  - Suppose $W = 20$ MHz, Noise figure = 2 dB
  - In dB: $P_w = N_0 + 10\log_{10} W = 10\log_{10}(kT) + NF + 10\log_{10} W = -174 + 2 + 73 = -99$ dBm
  - This is a very small number! Thermal noise is $= 10^{-9.9}$ mW $\approx$ 1 pW

Signal To Noise Ratio

- Complex baseband signal is $y(t) = y_0(t) + w(t)$



- Signal to Noise Ratio: Key ratio in communications:
  - In linear scale

$$SNR = \frac{\text{Signal Power}}{\text{Noise power}} = \frac{P_0}{P_w}$$

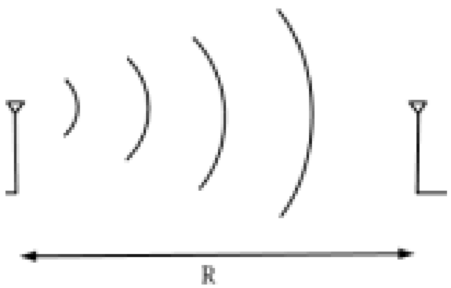
  - Often in dB:

$$SNR[dB] = P_0[dBm] - P_w[dBm]$$

  - Note the units
- Describes relative strength of signal to noise

Example: SNR of a Wireless Signal

- Freespace path loss from Friis' Law
  - P_r, P_t: Transmit and receive power
  - G_r, G_t: Antenna gains due to directivity
  - f_c: Carrier frequency, c: speed of light
  - d: TX-RX separation

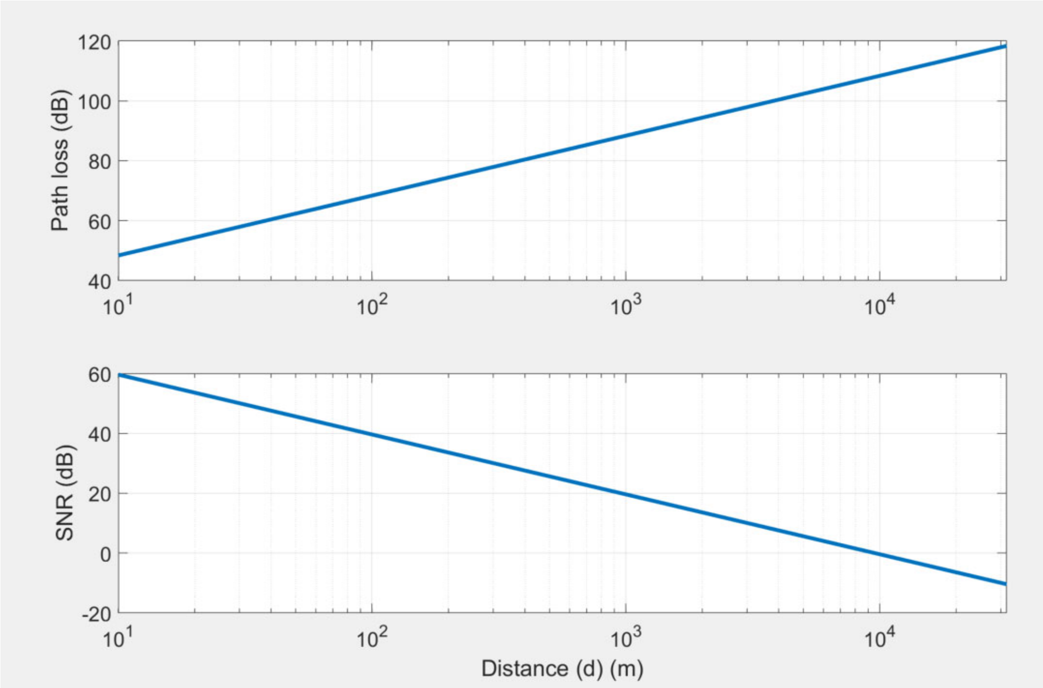


- Hence SNR at distance d is: $SNR = \frac{P_r}{N_0 W} = \frac{P_t G_t G_r}{N_0 W}\left(\frac{c}{4\pi d f_c}\right)^2$
- In dB:

$$SNR\ [dB] = P_t\ + G_t + G_r - kT - NF - 10\log_{10}(W) + 20\log_{10}\left(\frac{c}{4\pi d f_c}\right)$$

Free-Space SNR Visualized

- Parameters:
  - $f_c = 28$ GHz
  - NF = 6 dB
  - $G_t = 21$ dBi, $G_r = 12$ dBi
  - $P_t = 30$ dBm
  - W = 1 GHz
- SNR = 0 dB as far away as 10 km!

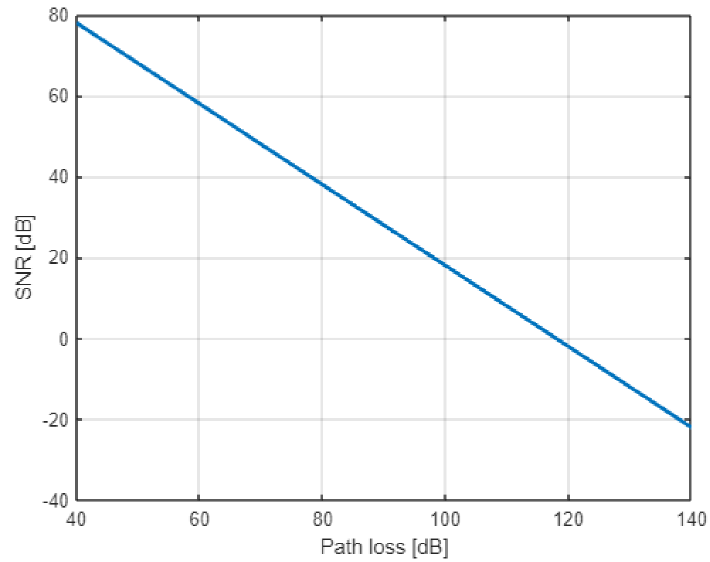


In Class Exercise

Computing SNR

In this example, we do a simple SNR calculation. Consider a system with the parameters below. For each path loss value, `PLTest(i)`, find the SNR value, SNR(i). Plot `SNR` vs. `PLTest`.

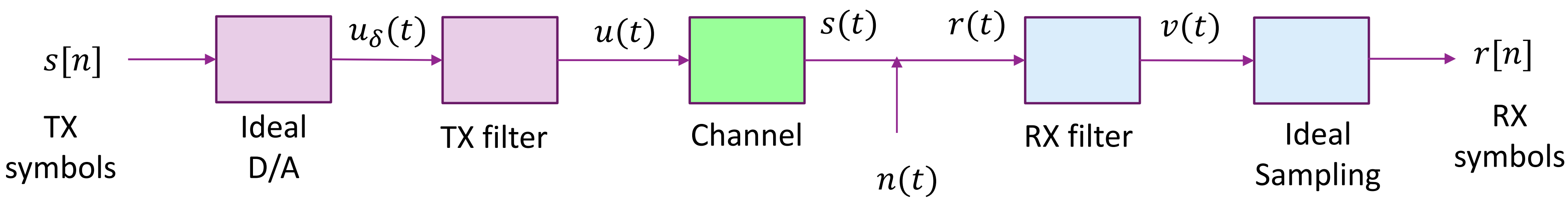
```
Ptx = 23;       % TX power in dBm
bw = 20e6;      % Bandwidth in MHz
PLTest = linspace(40,140,100)';  % Path loss values
NF = 6;    % RX noise figure in dB
```



Outline

- Passband and baseband noise, signal to noise ratio
- Noise in the discrete symbols
- ML Detection
- Symbol detection
- Probability of error

End-to-End System So Far



- Assume that noise $n(t)$ is complex AWGN
- What is the effect of noise on the received symbols?

Signal and Noise Components

- Received baseband signal: $r(t) = s(t) + n(t)$
  - $r(t), s(t)$: RX and TX complex baseband signals
  - $n(t)$ complex WGN noise with PSD N_0
- Receiver performs two steps:
  - Filtering: $v(t) = p_{rx}(t) * r(t)$
  - Sampling: $r[n] = v(nT)$
- Using linearity, spilt $r[n]$ into two components: $r[n] = r_0[n] + w[n]$
  - $r_0[n] =$ component due to signal $s(t)$
  - $w[n]$ = component due to noise
- From previous lecture, $r_0[n] = h[n] * s[n]$, $h[n]$ = effective discrete-time channel
- What is $w[n]$?

Noise Component

- Noise: $n(t)$ is complex WGN, PSD=N_0
- Analyze noise through the two receiver stages:
  - Filtering: $v_{noise}(t) = p_{rx}(t) * n(t)$
  - Sampling: $w[n] = v_{noise}(nT)$
- Each noise sample is given by convolution:

$$w[n] = \int n(t) p_{rx}(nT - t) dt = \int n(t) \phi_n^*(t) dt, \qquad \phi_n(t) := p_{rx}^*(nT - t)$$

- Theorem: Each sample $w[n]$ is complex Gaussian with $w[n] \sim CN(0, \sigma^2)$
  - Noise variance $\sigma^2 = \|p_{rx}\|^2 N_0$
  - Proof on board

Symbol Noise with Orthonormal RX Filtering

- Suppose that $\phi_n(t) \coloneqq p_{rx}^*(nT - t)$ is an orthonormal basis
- Theorem: Then $w[n] \sim CN(0, N_0)$ and the noise samples are independent
- Proof on board

Single Path Channel Model

- Simple model
  - Orthonormal modulation: $\phi_n(t) = p_{tx}(t - nT)$ is an orthonormal basis
  - Single path channel: $s(t) = hu(t - \tau)$
  - Matched filter receiver: $p_{rx}(t) = p_{tx}^*(-t)$
  - AWGN noise: $n(t)$ has PSD N_0

- Equivalent discrete-time model:

$$r[n] = hs[n] + w[n]$$

Power and Energy

- Equivalent discrete-time model: $r[n] = hs[n] + w[n], w[n] \sim CN(0, N_0)$
- Transmitted energy per symbol: $E_{tx} = E|s[n]|^2$
- Transmitted power: $P_{tx} = E_{tx}/T$
- Received energy per symbol: $E_{rx} = |h|^2 E_{tx}$
- Noise energy per symbol: N_0

- Path loss (in dB) = $-10 \log_{10} |h|^2 = 10 \log_{10} \frac{E_{tx}}{E_{rx}}$
  - Note the negative sign

Units

- E_{tx}, E_{rx} = Energy. Units are Joules in linear scale
  - Or dBJ / dBmJ in log scale
- P_{tx}, P_{rx} = Power. Units are Watts = Joules / sec.
  - Or dBm / dBW in log scale
- Noise energy N_0 has two equivalent units:
  - N_0 is in Joules: Represents noise energy per orthogonal sample
  - N_0 is in Watts / Hz: Represents noise power spectral density

Sample Question

- A transmitter sends symbols at a rate of 20 Msym/s and TX power of 23 dBm.
- What is the TX energy per symbol?
- Suppose that the path loss is 100 dB, what is the received symbol energy?
  - Note this is a very small amount of energy!
- Suppose that the receiver has a noise figure of 4 dB. What is the noise, N_0
- What is the signal-to-noise ratio E_{rx}/N_0 ?

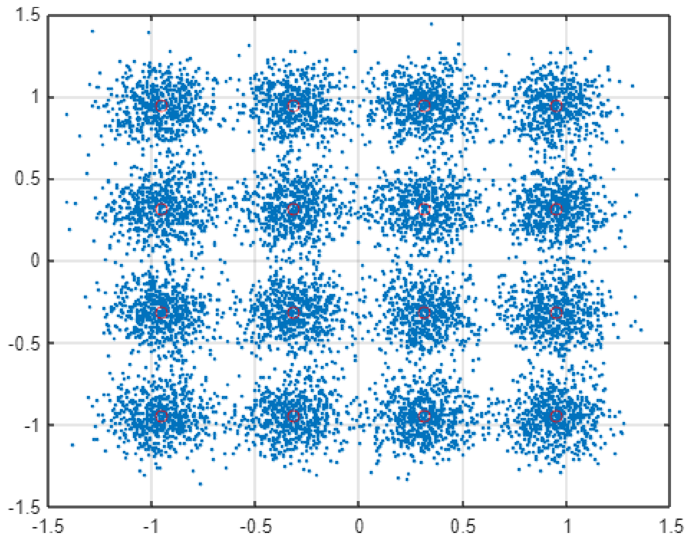
- Solution on board

In Class Exercise

Simulating Noise in Discrete Symbols

In this exercise, we will show how to add noise to the QAM symbols. First, create a set of random QAM modulated symbols with the following parameters.

```
nsym = 1e4;     % number of symbols
bitsPerSym = 4;  % bits per symbol
nbits = bitsPerSym * nsym;
```



Outline

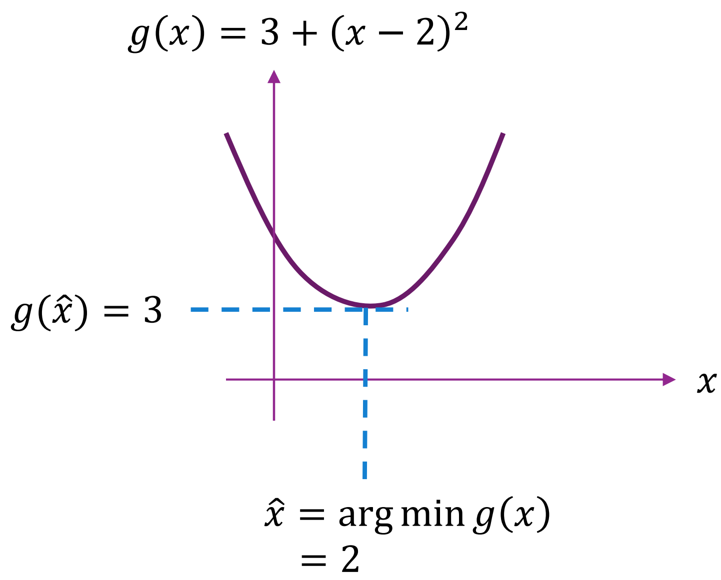
- Passband and baseband noise, signal to noise ratio
- Noise in the discrete symbols
- ML Detection
- Symbol detection
- Probability of error

Detection Theory

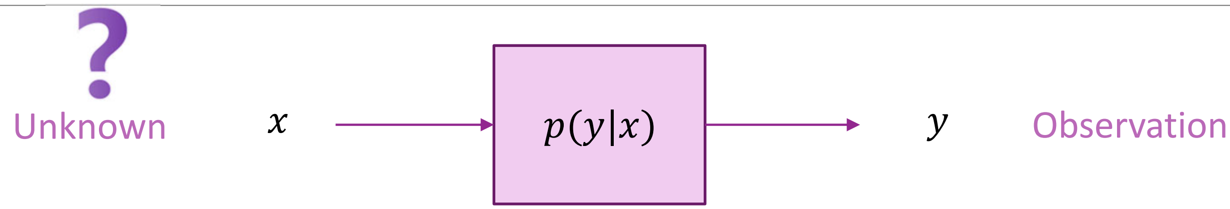
- Problem: Estimate some variable x from measurement y
- Basic problem in communications:
  - Detect a transmitted bit from a received symbol
  - Detect if a transmission occurred
  - Estimate a channel parameter
  - …
- And in many other fields:
  - Pattern recognition, image recognition, speech recognition
  - Machine learning: Estimate parameters in a model
  - …

Min and Arg Min

- Given a function $g(x)$
- $\min_x g(x)$ = minimum value of function
- $\arg\min_x g(x)$ = value of x that achieves the minimum
- Example: $g(x) = 3 + (x-2)^2$
  - Function achieves min $g(x) = 3$ at $x = 2$
  - $\min_x g(x) = 3, \arg\min_x g(x) = 2$
- May also restrict to a domain
  - $\arg\max_{x \in A} g(x)$ = maximum input restricted to a set A



Maximum Likelihood Estimation



- Statistical view: Model observation y as a random function of unknown x
  - x may be random or deterministic
- Describe by likelihood function $p(y|x)$
  - Conditional probability of y given measurements x
- Maximum likelihood principle:
  - Select variable x that is most likely

$$\hat{x} = \arg\max_{x} p(y|x)$$

Likelihood Ratio

- Consider binary detection case: $x \in \{0,1\}$
  - Two possible choices for unknown
- We have two likelihoods: $p(y|x = 0)$ and $p(y|x = 1)$
- Log likelihood ratio:

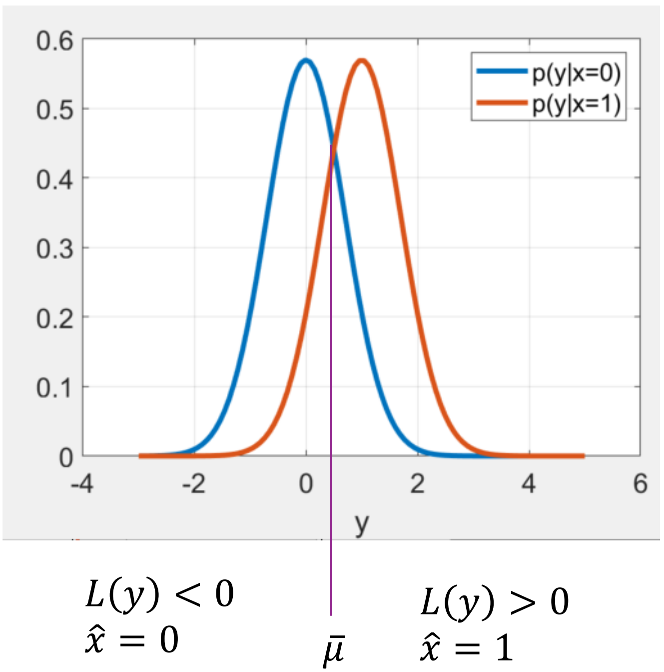
$$L(y) := \ln \frac{p(y|x = 1)}{p(y|x = 0)}$$

- ML estimation selects:

$$\hat{x} = \begin{cases} 1 & \text{if } L(x) \geq 0 \\ 0 & \text{if } L(x) \leq 0 \end{cases}$$

Example: Two Gaussians, Different Means

- Consider binary classification: $x = 0,1$
  - $p(y|x=j) = N\left(y\middle|\mu_j, \sigma^2\right), \mu_1 > \mu_0$
  - Two Gaussians with same variance
- Likelihood:
  - $p(y|x=j) = \frac{1}{\sqrt{2\pi}\sigma}\exp(-\frac{1}{2\sigma^2}(y-\mu_i)^2)$
  - $L(y) \coloneqq \ln\frac{p(y|1)}{p(y|0)} = -\frac{1}{2\sigma^2}[(y-\mu_1)^2 - (y-\mu_0)^2]$
  - With some algebra: $L(y) = \frac{(\mu_1-\mu_0)}{\sigma^2}[y-\bar{\mu}], \bar{\mu} = \frac{\mu_0+\mu_1}{2}$
- ML estimate:
  - $\hat{y} = 1 \Leftrightarrow L(y) \geq 0 \Leftrightarrow y \geq \bar{\mu}$
  - With some algebra we get: $\hat{x} = \begin{cases} 1 & \text{if } y > \bar{\mu} \\ 0 & \text{if } y \leq \bar{\mu}\end{cases}$,

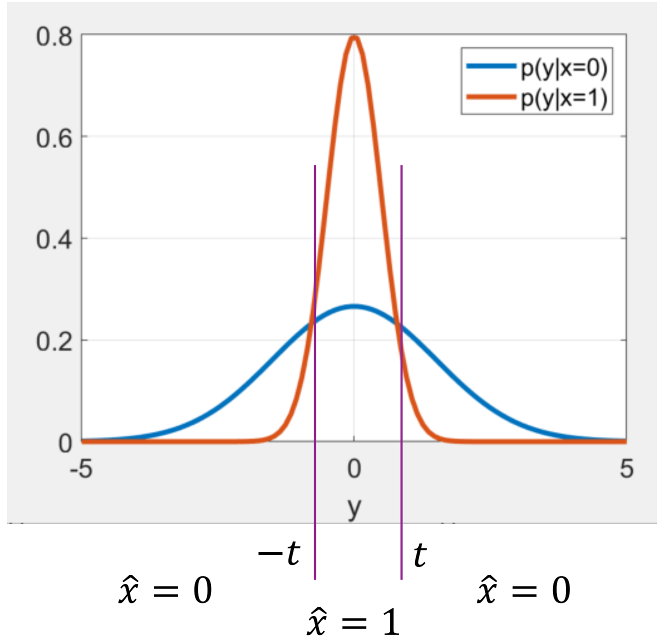


Example: Two Gaussians, Different Variances

- Consider binary classification: $x = 0,1$
  - $p(y|x=j) = N\left(y \middle| 0, \sigma_j^2\right), \sigma_1 > \sigma_0$
  - Two Gaussians with different variances

- Likelihood:
  - $p(y|x=j) = \frac{1}{\sqrt{2\pi}\sigma_j} \exp(-\frac{y^2}{2\sigma_j^2})$
  - $L(y) \coloneqq \ln \frac{p(y|1)}{p(y|0)} = \frac{y^2}{2}\left[\frac{1}{\sigma_0^2} - \frac{1}{\sigma_1^2}\right] - \frac{1}{2}\ln(\frac{\sigma_1^2}{\sigma_0^2})$

- ML estimate:
  - $\hat{y} = 1 \Leftrightarrow L(y) \geq 0 \Leftrightarrow |y| \geq t$
  - $t = \left[\frac{1}{\sigma_0^2} - \frac{1}{\sigma_1^2}\right]^{-1} \ln(\frac{\sigma_1^2}{\sigma_0^2})$



Outline

- ❑Passband and baseband noise, signal to noise ratio
- ❑Noise in the discrete symbols
- ❑ML Detection
- ❑Symbol detection
- ❑Probability of error

Demodulation

- Discrete-time model: $r[n] = hs[n] + w[n],\ w[n] = CN(0, N_0)$
- Suppose receiver knows:
  - $r[n]$ = received symbol
  - h = channel gain (it learns this through channel estimation from other symbols. Not covered here)
  - $s[n] \in \{s_1, \dots, s_M\}$ constellation set.
- Demodulation problem: Estimate which symbol $s[n] \in \{s_1, \dots, s_M\}$ was transmitted.

ML Estimation for Symbol Demodulation

- Demodulation problem: $r = hs + w,\ w \sim CN(0, N_0),\ s \in \{s_1, \dots, s_M\}$
  - Drop the sample index n
- Maximum likelihood estimation:

$$\hat{s} = \arg \max_{s=s_1,\dots s_M} p(r|s = s_m)$$

- Given s and h: $r \sim CN(hs, N_0)$
- Hence,

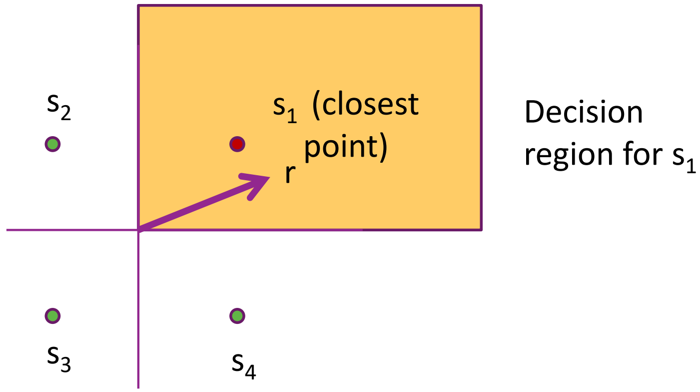
$$p(r|s) = \frac{1}{\pi\, N_0} \exp\left(-\frac{|r - hs|^2}{N_0}\right)$$

Nearest Symbol Detection

- Likelihood: $p(r|s) = \frac{1}{\pi N_0} \exp\left(-\frac{|r-hs|^2}{N_0}\right)$
- MLE is: $\hat{s} = \arg\max_s p(r|s) = \arg\min_s |r - hs|^2 = \arg\min_s |z - s|^2$
- Here, $z = \frac{r}{h}$ = equalized symbol.

- Procedure:
  - Step 1: Equalize the symbol: $z = \frac{r}{h}$
  - Step 2: Find $s = s_1, \ldots, s_M$ closest to z in complex plane

Decision Regions



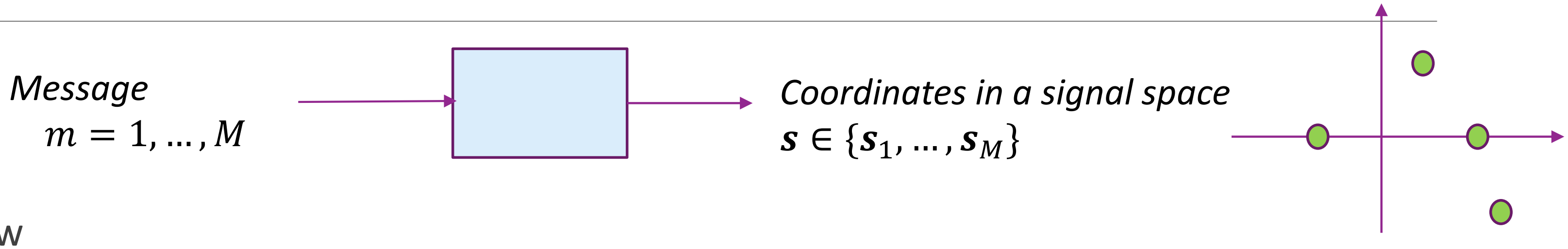
Example: Decision region in QPSK

- ML estimate is closest point in constellation to z: $\hat{s} = \arg\min_i \|z - s_i\|$
- Decision region for a point s_m:
  - set of points r where s_m is the closest point: $D_m = \{\boldsymbol{r} | \hat{s} = s_m\}$

Sample Problems

- Draw decision regions for:
  - QPSK
  - 16-QAM
  - 8-PSK
  - General constellations

Detection in a General Signal Space



- Signal space view
  - Input is a message $m = 1, \ldots, M$
  - Each output has a coordinate vector $\boldsymbol{s}_1, \ldots, \boldsymbol{s}_M \in \mathbb{F}^N$
- Suppose receive $\boldsymbol{r} = \boldsymbol{s}_m + \boldsymbol{w}, \quad \boldsymbol{w} \sim CN(\boldsymbol{0}, \sigma^2 \boldsymbol{I})$
  - Noise is independent and Gaussian in each symbol
- Theorem: The ML detector for the general signal space is:

$$\hat{\boldsymbol{s}} = \arg\min_{\boldsymbol{s}} \|\boldsymbol{r} - \boldsymbol{s}\|^2$$

  - Proof on next slide
- Consequence: Finds the closest vector in the N-dimensional space

Detection in a General Signal Space

- Proof of Theorem:
  - Given $\boldsymbol{s}$, each component r_n is independent with $r_n = s_n + w_n$
  - Therefore, $r_n \sim CN(s_n, N_0)$
  - Therefore, $p(r_n|s_n) = \frac{1}{\pi N_0} \exp\left(-\frac{1}{N_0}|r_n - s_n|^2\right)$
  - Since the components are independent:

$$p(\boldsymbol{r}|\boldsymbol{s}) = \prod_n p(r_n|s_n) = \frac{1}{(\pi N_0)^N} \prod_n \exp\left(-\frac{1}{N_0}|r_n - s_n|^2\right)$$
$$= \frac{1}{(\pi N_0)^N} \exp\left(-\frac{1}{N_0}\sum_n |r_n - s_n|^2\right) = \frac{1}{(\pi N_0)^N} \exp\left(-\frac{1}{N_0}\|\boldsymbol{r} - \boldsymbol{s}\|^2\right)$$

  - Hence, ML detector is:

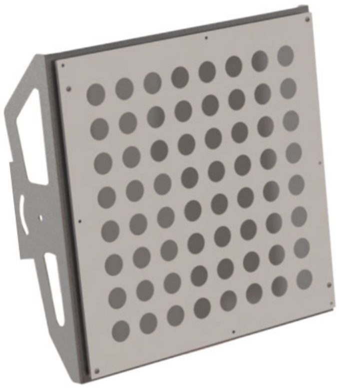
$$\hat{\boldsymbol{s}} = \arg\max_{\mathbf{s}} p(\boldsymbol{r}|\boldsymbol{s}) = \arg\min_{\mathbf{s}} \|\boldsymbol{r} - \boldsymbol{s}\|^2$$

Example: Multiple Measurements

- Transmit a single symbol: $x \in \{x_1, \ldots, x_M\} \in \mathbb{C}$
- Receive multiple measurements:

$$r[n] = h[n]x + w[n], \qquad n = 0, \ldots, N-1$$

- Same symbol x is transmitted over multiple samples
- Multiple samples can arise in many scenarios:
  - Different time samples
  - Samples from different antennas



Ex: 5.6GHz Massive MIMO array
The received signal is a vector

- $r[n]$= signal to antenna element n
- $h[n]$=channel from TX to the element

Example: Multiple Measurements

- Receive multiple measurements: $r[n] = h[n]x + w[n], \quad n = 0, \ldots, N-1$
- In vector form: $\boldsymbol{r} = \boldsymbol{h}x + \boldsymbol{w}$
- Each transmitted signal is received as $\boldsymbol{s} = \boldsymbol{h}x.$ ML detector: $\hat{x} = \arg\min_{\boldsymbol{s}} \|\boldsymbol{r} - \boldsymbol{h}x\|^2$
- But, $\|\boldsymbol{r} - \boldsymbol{h}x\|^2 = \|\boldsymbol{r}\|^2 - 2Re(\boldsymbol{r}^*\boldsymbol{h}x) + |x|\text{^}2\|\boldsymbol{h}\|^2$
- Let $z = \frac{\boldsymbol{r}^*\boldsymbol{h}}{\|\boldsymbol{h}\|^2}$. This is called the equalized symbol.
- Then: $\|\boldsymbol{r} - \boldsymbol{h}x\|^2 = \|\boldsymbol{h}\|^2|z - x|^2 + \|\boldsymbol{r}\|^2 - \frac{|\boldsymbol{r}^*\boldsymbol{h}|^2}{\|\boldsymbol{h}\|^2}$
- Hence: $\hat{x} = \arg\min_{\boldsymbol{s}} \|\boldsymbol{r} - \boldsymbol{h}x\|^2 = \arg\min_{\mathrm{x}} |z - x|^2$
- Conclusion: Given multiple measurements:
  - Compute equalized symbol $z = \frac{\boldsymbol{r}^*\boldsymbol{h}}{\|\boldsymbol{h}\|^2}$
  - Demodulate from the received scalar symbol: $\hat{x} = \arg\min_{\mathrm{x}} |z - x|^2$

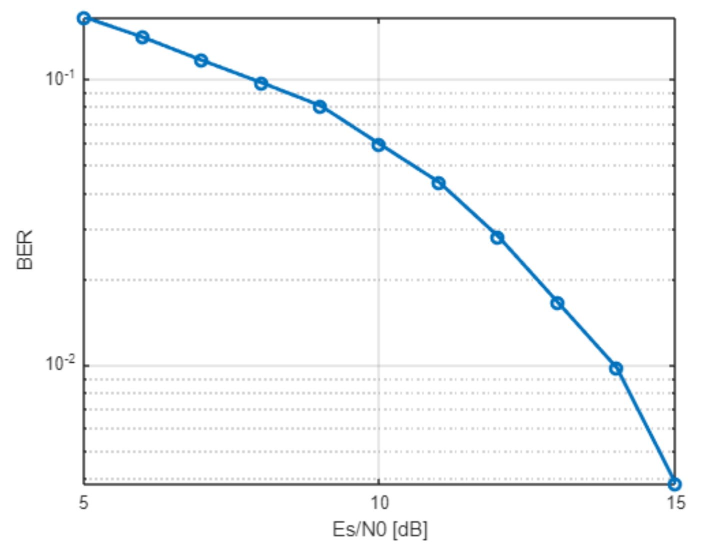
In Class Exercise

Measuring the BER

MATLAB's commnuication toolbox has an excellent command, qamdemod, that can be used to demodulate the bits from the noisy symbols. The syntax we will use is:

```
bhat = qamdemod(r,M,'OutputType','bit', 'UnitAveragePower', true);
```

Note that you have to provide the average power since it is the reference level. Use this command to get the estimated bits and find the BER. At Es/N0 = 15 dB the BER should be ~ 0.45%.



Outline

- Passband and baseband noise, signal to noise ratio
- Noise in the discrete symbols
- ML Detection
- Symbol detection
- Probability of error

Symbol Error Probability

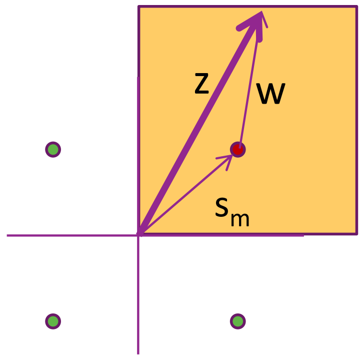
- Want to compute symbol error rate

$$SER = P(m \neq \hat{m})$$

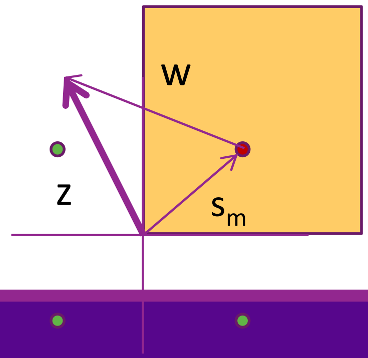
- Assume all constellation points equally likely
- Average SER:

$$SER = \frac{1}{M}\sum_{m=1}^{M} P(\hat{s} \neq s_m | s = s_m)$$

$$= \frac{1}{M}\sum_{m=1}^{M} P(z \notin D_m | s = s_m)$$



No error
z in correct decision region



Error
z not in
correct decision region

Signal to Noise Ratio

- Discrete-symbol model (no channel gain):

$$r = s + w, \qquad w \sim CN(0, N_0), \qquad s = s_1, \dots, s_M$$

- Received symbol energy: $E_s = \frac{1}{M} \sum_{m=1}^{M} |s_m|^2$
- Signal to noise ratio:

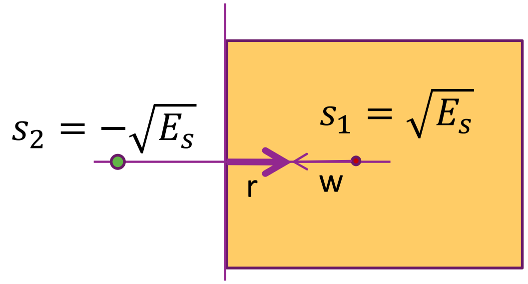
$$\gamma_s = \frac{E_s}{N_0}$$

  - Sometimes called SNR per symbol
- When there is a channel gain, $r = hs + w$. Replace E_s with $|h|^2 E_s$

SER for BPSK

- BPSK constellation: $s = \pm\sqrt{E_s}$
- AWGN channel:

$$r = s_i + n, \qquad n \sim CN(0, N_0)$$



- SER: By symmetry

$$\mathrm{SER} = P(\hat{m} = 2 | m = 1)$$

- Will show on board:

$$SER = Q\left(\sqrt{2\gamma_s}\right)$$

  - $\gamma_s = E_s/N_0$ symbol SNR
- Also, for BPSK:

$$\gamma_b = E_b/N_0 = \gamma_s$$

SER for QPSK

- SER for QPSK (will show on board)

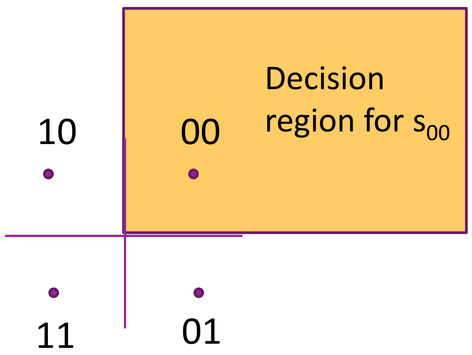
$$SER = 1 - \left(1 - Q(\sqrt{\gamma_s})\right)^2 = 2Q(\sqrt{\gamma_s}) - Q^2(\sqrt{\gamma_s})$$

- Look at SNR per bit
- High SNR asymptotic
- Compare to BPSK

$$s_m = \sqrt{\frac{E_s}{2}}(\pm 1 \pm i)$$

$$d_{\min} = \sqrt{2E_s}$$

QPSK or 4-QAM
2 bits / symbol
Smaller dmin

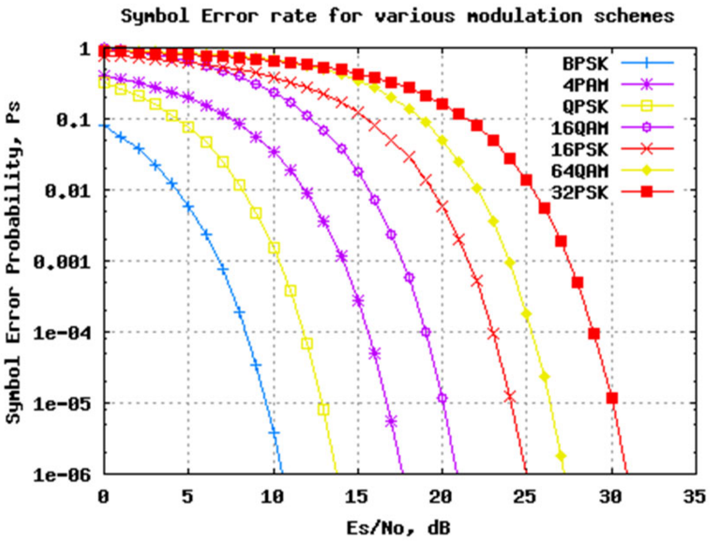


More Calculations

- If you are interested, Proakis “Digital Communications” derives error rates for many constellation types:
  - M-PSK, M-QAM, DQPSK, …
  - Provides exact formulae and various bounds

SER for Various Modulation Schemes

- Some observations:
  - QPSK has roughly same BER as BPSK for same Eb/N0
    - Note that SNR is shown in figure asEs/N0 not Eb/N0
  - M= QAM requires roughly 6 dB per bit above M=4
  - M-PSK is significantly less efficient that M-QAM



In Class Exercise

Comparing to the Theoretical Value

We can compare the BER we just simulated to the theoretical values. Proakis and other references show that, at high SNR, the symbol error rate is approximately:

$$P_s \approx 2 \cdot Q\left(\sqrt{\frac{2E_b}{N_0}}\right) \quad \text{(4-QAM)}$$

$$P_s \approx 3 \cdot Q\left(\sqrt{\frac{4E_b}{5N_0}}\right) \quad \text{(16-QAM)}$$

where `Eb/N0` is in linear scale. With Gray coding, most symbol errors result in only one bit error per symbol. Hence at hig approximately:

```
ber ~= ser / bitsPerSym
```

Compute the theoretical BER for the Es/N0 values in `EsN0Theory`. Plot `berTheory` vs. `EsN0Theory` on the same graph a: You should get a good match.