STANFORD

*Lecture 5*

# Adapting Modulation Coding Scheme

*April 16, 2024*

**JOHN M. CIOFFI**

Hitachi Professor Emeritus of Engineering

Instructor EE379B – Spring 2024

# Announcements & Agenda

- Announcements
  - Problem Set #1 Solutions posted – see PS1 (also link at course page and connects to canvas where actual solution is stored)
  - Problem Set #2 Wed April 19 at 17:00
  - Sections **1.6**, 4.4
  - Problem Set #3 due April 26 at 17:00

- Agenda
  - Loading with Statistical Channels
  - Ergodic Coded-OFDM Loading
  - Spatial Modulation

- Problem Set 3 = PS3, due 4/26
  1. 4.13 basic C-OFDM design
  2. 4.14 ergodic water-fill
  3. 4.22 wireless spatial loading
  4. 4.16 estimating gain distribution
  5. 4.15 Simple Wi-Fi Loading

# Loading with Statistical Channels

***Subsection 4.4.2***

*PS3.1 (Prob 4.13), PS3.3 (Prob 4.22) and PS3.4 (Prob 4.16)*

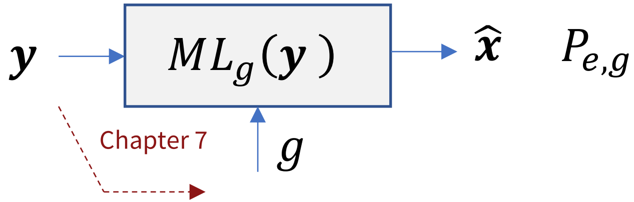
# The random-gain AWGN

- See 379A-L5:6-18 for statistical channels.
- The AWGNs (tones/space) have a gain parameter $g$ , so $p_{[\boldsymbol{y}\; g]/\boldsymbol{x}}$ where $g$ is random.
  - The gain is $g = \frac{|h|^2}{\sigma^2}$ for the AWGN, or more with variable gains/dimension $g_n \rightarrow {}^{SNR_{geo}}/_{\bar{\mathcal{E}}_x}$ , following Sep Th^m.
  - Effectively, each dimension has random $SNR_n = \mathcal{E}_n \cdot g_n$ where the designer can choose $\mathcal{E}_n$.

$$p_{[\boldsymbol{y}\; g]/\boldsymbol{x}} = p_{\boldsymbol{y}/[\boldsymbol{x}\; g]} \cdot \underbrace{p_{g/\boldsymbol{x}}}_{p_g}$$

$\boldsymbol{x}$ and $g$ are independent

- The ML/MAP receiver is a function of $g$.
  - Has error-probability distribution $P_{e,g}$ , $\langle P_e \rangle \triangleq \mathbb{E}\left[P_{e,g}\right]$.




- For Rayleigh

$$\langle P_e \rangle = \frac{1}{2} \cdot \left(1 - \sqrt{\frac{\kappa \cdot SNR}{\kappa \cdot SNR + 1}}\right) \cong \frac{1}{4\kappa \cdot SNR}$$

For large $SNR$

$\kappa = \frac{3}{M-1}$ for square QAM

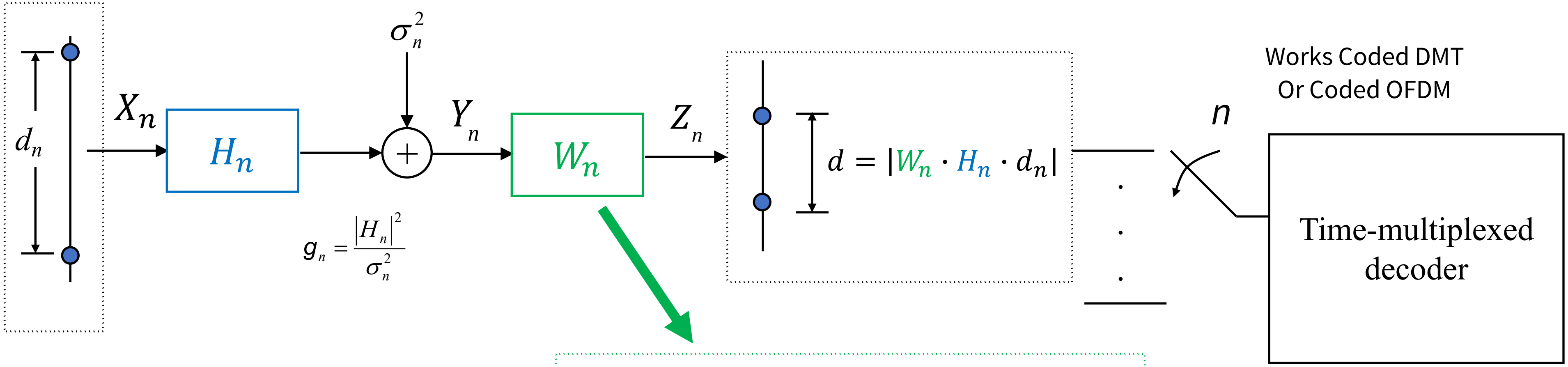
# Example MCS loading using Sep Thm

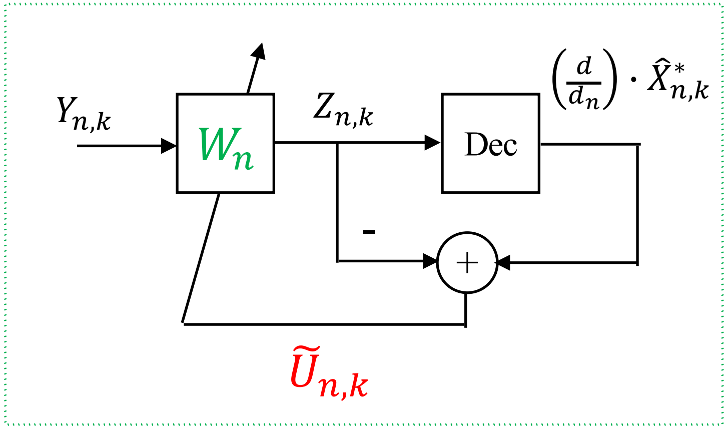
- Base code is the $r = \frac{1}{2}$, $d_{free} = 10$, 64-state code from Wi-Fi (see 379A):

- 48 used tones (20 MHz-wide channel); $SNR_{geo} = 14.5$ dB (computed with 0 dB gap) and $\mathcal{E}_n \equiv \mathcal{E}$.
  - Puncturing options
    - No puncturing $r = \frac{1}{2}$, $d_{free} = 10$
    - Puncture 3 bits from 12, $r = \frac{2}{3}$, $d_{free} = 6$
    - Pucture 4 bits from 12, $r = \frac{3}{4}$, $d_{free} = 5$
  - MCS Options
    - Uncoded 4QAM – nominally $P_e \cong 10^{-7}$ for a single AWGN; however, Sep Thm only applies with good codes – not viable.
    - Coded 16QAM at rate 2/3 has coding gain of 6 dB (although increase to 16QAM constellation costs 7 dB), but still able to get $P_e \cong 10^{-6}$ (Gray coding, BICM).
    - Coded 16QAM at rate ¾ has slightly higher gain, so also would work.
    - The 16QAM system thus has $b=\frac{3}{4}*2$=1.5 bits/dimension so indeed better rate than uncoded 4 QAM also.

- This presumes the $SNR_{geo} = 14.5$ dB remains constant.
  - Each use may need a new MCS solution when there is time-varying fading, which means $SNR_{geo}$ varies.
  - Wi-Fi trains on each channel use (so when channel is available and there is data to send).
  - Even video streams today are divided into packets and not sent as continuous stream.

# Multichannel Normalizer (or "FEQ"), See L4:16




## Zero-Forcing Algorithm

$$W_{n,k+1} = W_{n,k} + \mu_n \cdot \tilde{U}_{n,k} \cdot \left(\frac{d}{d_n}\right) \cdot \hat{X}^*_{n,k}$$

$$\tilde{\sigma}^2_{n,k+1} = (1 - \mu') \cdot \tilde{\sigma}^2_{n,k} + \mu' \cdot \left|\tilde{U}_{n,k}\right|^2$$

## Channel estimate

$$\hat{H}_n = \frac{d}{d_n \cdot W_n}$$

## Noise Estimate

$$\hat{\sigma}_n^2 = \frac{\tilde{\sigma}_n^2}{|W_n|^2}$$

**Rcvr can also estimate actual gain distribution $p_g$ from this.**

# Calculation of average and outage error probs

**Definition 4.4.4 [Outage Probability]** *The* **outage probability** *differs from the* **random-error probability** *according to (they differ in the sum's value range)*

$$\langle \bar{P}_e \rangle = \sum_{g > g_{out}} p_g \cdot \bar{N}_e \cdot Q\left[\sqrt{\frac{3 \cdot \bar{\mathcal{E}}_{\boldsymbol{x}} \cdot g \cdot d_{free}(r)}{|C| - 1}}\right] \quad (4.151)$$

$$\bar{P}_{out} = \sum_{g \leq g_{out}} p_g \ , \ \textit{respectively,} \quad (4.152)$$

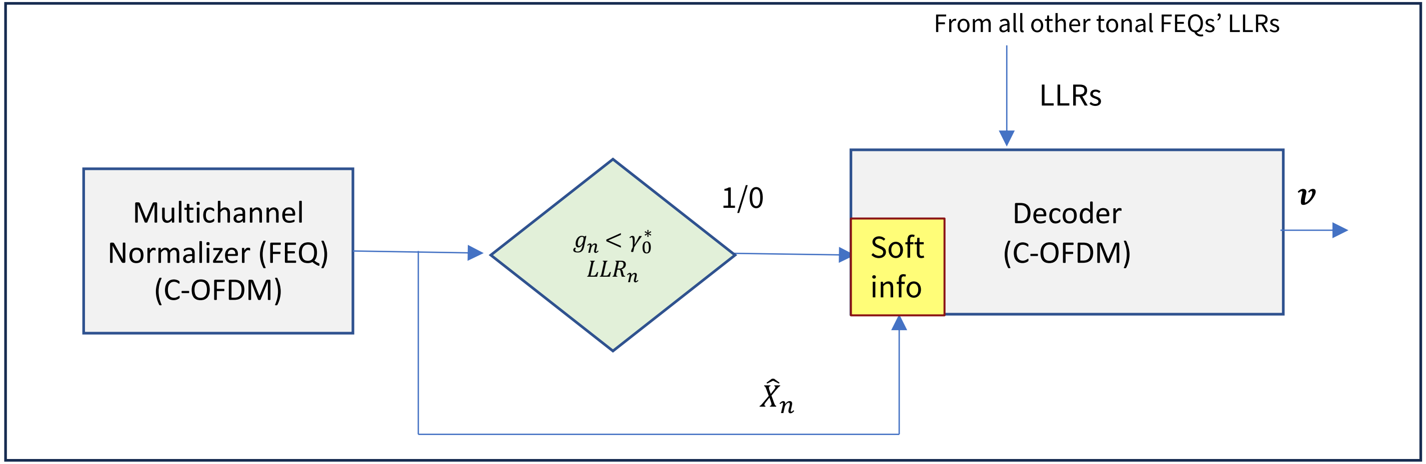
*where $g_{out}$ is a threshold channel SNR to be determined so that (4.151) holds, while the "outage" corresponding to lower gains (meaning very poor performance with high error probability) must be accommodated by the receiver's erasure marking in decoding. The fraction $\frac{3}{|C|-1}$ can be adjusted to $\kappa$ if the design uses non-square constellations, but the concept is the same.*

In general, soft decode.

- Average error prob and outage depend on a channel-gain distribution $p_g$ that is given or measured (by receiver).
- The average error probability is weighted by $p_g$ for just those transmissions not in outage.
- **Quasi-stationary assumption** is that the gain $g_n$ is constant for each decoder use (over the codeword) – it can vary over $n$ , but not time.
  - Think tone for multicarrier – what this really means is $SNR_{geo}$ is constant over each decoder use of tones/spatial dimensions in the context of separation theorem.
  - That is, coherence **time** is best longer than a codeword.
  - If not, hard-decoding applies at lower data rate with that $\langle P_b \rangle^{\left\lfloor \frac{d_{free}+1}{2} \right\rfloor}$ for a BSC.

# LLRs generally with FEQ & Decoder

- FEQ provides LL/soft information to the decoder.




- This works if code is good $\Gamma \to 0$ dB;
- Even works well if erasures used, then hard (e.g., Reed Solomon) code recovers if < P/2 erred ss.
  - Upcoming Ergodic Water-Fill (both MA and RA) provide theoretical guidance, but not practical.
  - $LLR_n$=0 if $g_n < \gamma_0^*$.
- **LL/Erasure avoids:**
  - "feedback delay that $g_n < \gamma_0$ , so don't transmit" would take too long (issue largely only in wireless)
  - On/off or variable gain that frustrates power amplifiers (on/off transients) and receiver automatic gain-control circuits
- Adaptive power is often feasible for space (MIMO) to use/not use certain spatial channels, but not for time-frequency.

# Ergodic rate-adaptive loading

**Definition 4.4.5** **[Ergodic Rate-Adaptive Coded MT Loading with constant energy]** *Ergodic Rate-adaptive coded MT loading* **with constant energy** *solves*

$$\textbf{objective:} \quad \max_{r,|C|,g_{out}} \quad r \cdot \log_2 |C| \tag{4.155}$$

$$\tag{4.156}$$

$$\textbf{subject to:} \quad \langle \bar{P}_e \rangle = \sum_{g > g_{out}} p_g \cdot \bar{N}_e \cdot Q\left[\sqrt{\frac{3 \cdot \bar{\mathcal{E}}_{\boldsymbol{x}} \cdot g \cdot d_{free}(r)}{|C|-1}}\right] \leq \bar{P}_e \tag{4.157}$$

$$r \quad \leq 1 - \sum_{g \leq g_{out}} p_g , \tag{4.158}$$

*where the algorithm selects the code rate $0 < r \leq 1$ from among the allowed code rates, and $|C|$ is the selected (usually square) QAM constellation size in (4.155) - (4.158). The outage threshold $g_{out}$ characterizes the two sums that are computed for each candidate ordered pair of $[r, |C|]$. The fraction $\frac{3}{|C|-1}$ can be adjusted to $\kappa$ with non-square constellations, but the concept is the same.*

**Q-func is correct with quasi-static assumption.**

**Also BICM is presumed so that this formula essentially assumes $\log_2|C|$ parallel bit channels.**

**Definition 4.4.3** **[Average Geometric Channel ratio]** *The* **average geometric channel gain** *is*

$$\gamma^*_{geo} \triangleq \prod_{g \in \mathcal{G}^*} (g)^{\left[\frac{p_g}{\sum_{g \in \mathcal{G}^*} p_g}\right]} . \tag{4.143}$$

- Finds the cut-off channel gain $g_0$ that maximizes data rate ($\tilde{b} = r \cdot |\mathcal{C}|$).
  - The $[r, |C|]$ choice is over allowed constellations and codes with ($d_{free}(r)$), for the average error probability.
  - Assumes $r \leq 1 - P_{out}$ . So applied code with rate $r$ can correct the subsymbols lost to outage (think RS can at least do this).

# Loading on Actual Wireless Channel with Flat Energy

- Compute gap-based data rate for both average and max (over all tones) with rough gap estimate.

$$b_{flat-geo} = \frac{1}{N} \cdot \sum_{n=1}^{N} \log_2 \left( 1 + \frac{\bar{\mathcal{E}}_x \cdot g_n}{\Gamma} \right) \qquad b_{max} = \log_2 (\text{max constellation size}) \leq \max_n \left\{ \log_2 \left( 1 + \frac{\bar{\mathcal{E}}_x \cdot g_n}{\Gamma} \right) \right\}$$

- Compute code rate for "good code" (low gap) with this flat energy.
  - Then this code rate $r$ then leads to a $d_{free}$ for the selected code.

$$r = \frac{b_{flat-geo}}{b_{max}} \leq 1$$

- Compute outage probability from $r$ as $\bar{P}_{out} = 1 - r$.

- Estimate $p_g$ by **binning** or counting of $g$ values in 6 dB – *10 log10 ($d_{free-new}$/$d_{free-old}$)*, over which the $b$ value does not change the constellation-size choice (see example L5:13).
  - Binning counts FEQ noise-estimate values in each range (current symbol or over many symbols).

- Solve for $g_0$ -- tones with $g<g_0$ will be "erased."

- Indicate "erasure" (delete from sum is one way) in ML detector.
  - This could be an erasure in Reed-Solomon.
  - It can also be zero LLR in iterative decoder for bits corresponding to that tone.

$$\bar{P}_{out} = \sum_{g<g_0} p_g$$

# Error Correction Code (Section 2.2)

- Example:
  - Use Reed Solomon Block FEC  byte wise (N <256):
    - Parity *P* is up to 32 bytes.
    - It corrects up to *P/2* bytes if in random codeword positions, or up to *P* if "erasures" (locations of likely byte error) are determined.
  - Suppose $P_{out}$ = 5% and *N=200,*
    - then 10 to 20 parity bytes are needed.
  - Suppose $P_{out}$ = 50%, then *N=40* and *P > 20* might be needed  (low rate code < ½)
- Designs can also use soft iterative decoding with BICM on binary code use.

# Wireless Example

- Extension of well-known code as example (64-state rate-1/2 code with puncturing) has:
  - >> r = 0.9 0.8 0.75 0.67 0.5 0.25 0.2
  - >> dfree = 2 4 6 7 10 20 25
- Given (measured) channel-gain distribution
  - >> g = 3 30 300 600 1200 2400 4800 10000
  - >> pg=[.11 .1 .03 .05 .35 .2 .1 .06];

```
>> prob2 = kron(ones(7,1),pg);
Pout=cumsum(prob2(1,1:8)) % =
   0.1100   0.2100   0.2400   0.2900   0.6400   0.8400   0.9400   1.0000

>> ones(1,8)-Pout =
   0.8900   0.7900   0.7600   0.7100   0.3600   0.1600   0.0600        0

>> r =
   0.9000   0.8000   0.7500   0.6700   0.5000   0.2500   0.2000
```

Not correctable    correctable    Not correctable

# Wireless Example continued for <Pe>

- The SNR (with $\tilde{\mathcal{E}}_x = 1$ ) is

```
>> SNR=kron(dfree',g);
>> 10*log10(SNR) =  (in dB)
```

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| 7.7815 | 17.7815 | 27.7815 | 30.7918 | 33.8021 | 36.8124 | 39.8227 | 43.0103 |
| 10.7918 | 20.7918 | 30.7918 | 33.8021 | 36.8124 | 39.8227 | 42.8330 | 46.0206 |
| 11.7609 | 21.7609 | 31.7609 | 34.7712 | 37.7815 | 40.7918 | 43.8021 | 46.9897 |
| 12.5527 | 22.5527 | 32.5527 | 35.5630 | 38.5733 | 41.5836 | 44.5939 | 47.7815 |
| 14.7712 | 24.7712 | 34.7712 | 37.7815 | 40.7918 | 43.8021 | 46.8124 | 50.0000 |
| 17.7815 | 27.7815 | 37.7815 | 40.7918 | 43.8021 | 46.8124 | 49.8227 | 53.0103 |
| 18.7506 | 28.7506 | 38.7506 | 41.7609 | 44.7712 | 47.7815 | 50.7918 | 53.9794 |

$\tilde{\mathcal{E}}_x \cdot d_{free} \cdot g$

**Only rows 3-4 are eligible.**

- Compute $\langle P_e \rangle$ for several SNRs and SQ QAM Constellations
  - 4QAM -- >> prob1=q(sqrt(SNR(1:4,1:4)))
  - 16 QAM -- >>prob1=2*q(sqrt((3/15)*SNR(1:4,1:4)))
  - etc

**Compute <Pe> tables**

## 4SQ QAM

$g_{out}$ →

```
>> >> prob1=qfunc(sqrt(SNR(3:4,:)))
prob1 =   1.0e-04 *
```

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| 0.1105 | 0.0000 | 0 | 0 | 0 | 0 | 0 | 0 |
| 0.0230 | 0.0000 | 0 | 0 | 0 | 0 | 0 | 0 |

```
>> avePe=cumsum((prob2(3:4,1:8).*prob1)','reverse')'
avePe =   1.0e-05 *
```

| r | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| .75 | 0.1215 | 0.0000 | 0 | 0 | 0 | 0 | 0 | 0 |
| .67 | 0.0253 | 0.0000 | 0 | 0 | 0 | 0 | 0 | 0 |

$\bar{b} = \frac{3}{4} \cdot 1 = .75$

## 16SQ QAM

```
>> prob1=qfunc((3/15)*sqrt(SNR(3:4,:))) % =
```

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| 0.1981 | 0.0036 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0 |
| 0.1797 | 0.0019 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0 |

```
>> 100*avePe=cumsum((prob2(3:4,1:8).*prob1)','reverse')' % =
```

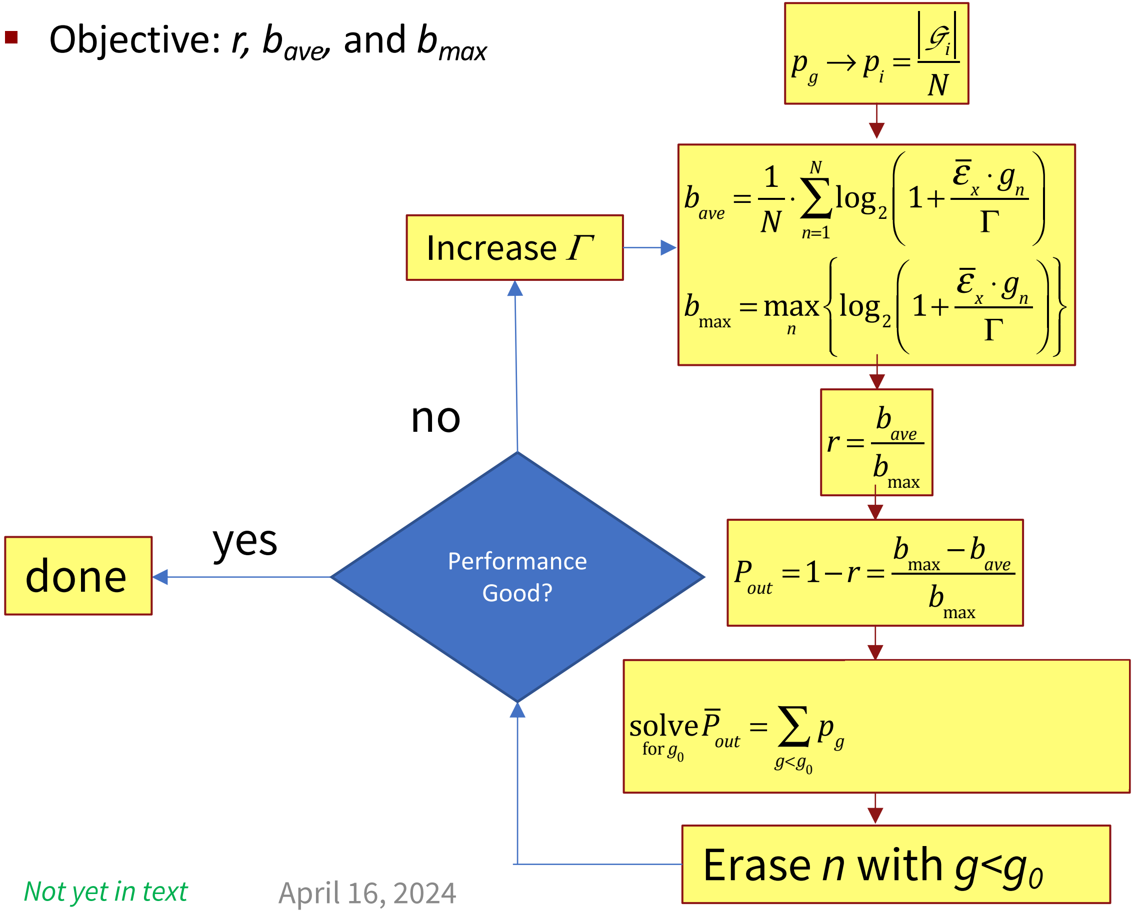
| | | | | | | | |
|---|---|---|---|---|---|---|---|
| 2.2152 | 0.0365 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0 |
| 1.9954 | 0.0188 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0 |

$\bar{b} = \frac{3}{4} \cdot 2 =$ 1.5 bits/dim

- 64 QAM doesn't quite work with anything, so 16SQ with r=3/4 is best design, see Section 4.4.2.3.

# Project: Flow Chart for C-OFDM Loading?

- Objective: $r$, $b_{ave}$, and $b_{max}$




Calculate Distribution

Compute averate and max bits/tone

For the used code find $r$ and $d_{free}$

Code should be able to correct for $P_{out}$ % erased

Need to approximate Q function or store it for each $i$ value

**untested by instructor**

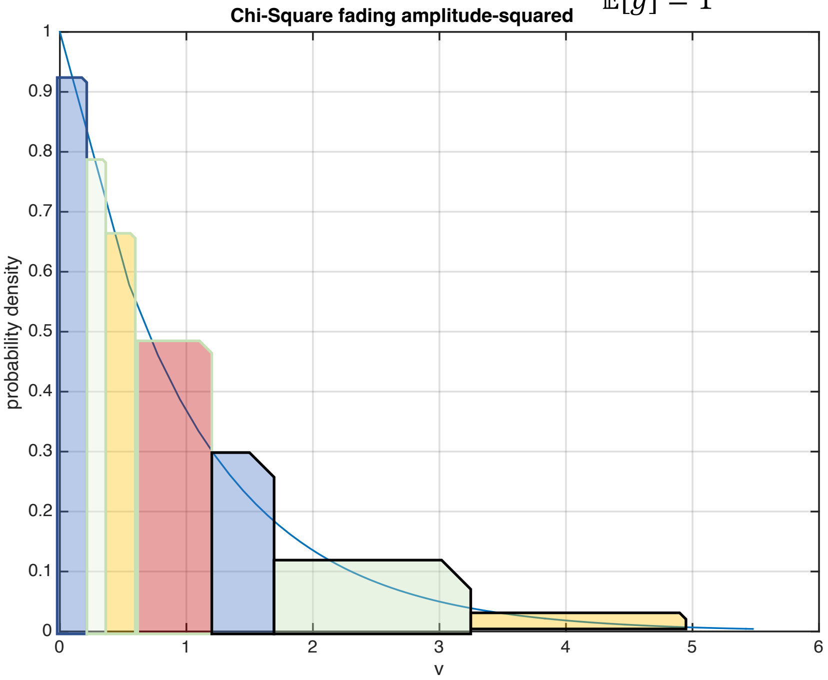
# Ergodic Coded-OFDM Loading

***Subsection 4.4.2***

*See PS3.2 (Prob 4.14)*

# Approximate the distribution as discrete

$\mathbb{E}[g] = 1$




$$p_g(v) \cdot dv \rightarrow p_{g,n}$$

May care about lower ranges more and so more finely divide distribution there into samples

- This simplifies calculations for loading, and the Rayleigh model was only approximate anyway.

# Discrete Distributions

- All distributions (Rayleigh, Ricean, Log-Normal, etc) are gross approximations in wireless.
- Approximate by discrete distributions (which can be learned): $p_g(v) \cdot dv \rightarrow p_{g,n}$
- May need to renormalize so probabilities sum to 1

$$p_{g,n} = \frac{p_g(n \cdot \Delta) \cdot \Delta}{\sum_n p_g(n \cdot \Delta) \cdot \Delta}$$

- The probability $p_{g,n}$ represents the fraction of time that $g_n - \frac{\Delta n}{2} < g \leq g_n + \frac{\Delta n}{2}$.
  - Intervals $\Delta_n$ do not need to be the same size (may want to align with constellation-size choices).

- Simplify so that $g$ is discrete set of center values $p_g\ (n\Delta)$ and discrete index $g \in \mathcal{G}$, size $|\mathcal{G}|$.
- $g$ takes place of dimension index, almost (weight not 1 as with integer index).

# Average bit rate and ergodic capacity ($\Gamma = 0$ dB)

- Average bit rate is: $$\langle b \rangle = \sum_{g \in \mathcal{G}} p_g \cdot \log_2\left(1 + \frac{\mathcal{E}_{x,g} \cdot g}{\Gamma}\right)$$

Ave Mutual Info has $\Gamma$=0 dB.

- Average energy is: $$\mathcal{E}_x = \sum_{g \in \mathcal{G}} \mathcal{E}_{x,g} \cdot p_g$$

**$g$ takes $n$'s place in DMT** RA.

See PS3.2 (Prob 4.14)

- Rate Adaptive and Margin-Adaptive Water-fill have $\mathcal{E}_g = K - \Gamma/g$.

$$\mathcal{E}_{x,g} = K_{ra} - \frac{\Gamma}{g} \quad ,$$

$$\begin{aligned} \mathcal{E}_{\boldsymbol{x}} &= \sum_{g \in \mathcal{G}^*} p_g \cdot \left(K_{ra} - \frac{\Gamma}{g}\right) \\ &= K_{ra} \cdot \sum_{g \in \mathcal{G}^*} p_g - \sum_{g \in \mathcal{G}^*} p_g \cdot \frac{\Gamma}{g} \\ K_{ra} &= \frac{\mathcal{E}_{\boldsymbol{x}} + \Gamma \cdot \sum_{g \in \mathcal{G}^*} \frac{p_g}{g}}{\sum_{g \in \mathcal{G}^*} p_g} \end{aligned}$$

**RA**

$$g_{geo}(*) \triangleq \prod_{g \in \mathcal{G}^*} (g)^{\left[\frac{p_g}{\sum_{g \in \mathcal{G}^*} p_g}\right]}$$

$$\begin{aligned} <b> &= \log_2 \prod_{g \in \mathcal{G}^*} \left(1 + \frac{\mathcal{E}_{x,g} \cdot g}{\Gamma}\right)^{p_g} \\ &= \log_2 \prod_{g \in \mathcal{G}^*} \left(\frac{K_{ma} \cdot g}{\Gamma}\right)^{p_g} \\ 2^{<b>} &= \left(\frac{K_{ma}}{\Gamma}\right)^{\sum_{g \in \mathcal{G}^*} p_g} \cdot \prod_{g \in \mathcal{G}^*} g^{p_g} \\ K_{ma} &= \Gamma \cdot \left(\frac{2^{<b>}}{\prod_{g \in \mathcal{G}^*} g^{p_g}}\right)^{\frac{1}{\sum_{g \in \mathcal{G}^*} p_g}} \\ &= \Gamma \cdot \frac{\left(2^{<b>}\right)^{\left[\frac{1}{\sum_{g \in \mathcal{G}^*} p_g}\right]}}{g^*_{geo}} \end{aligned}$$

**MA**

# Ergodic Water-filling - Goldsmith

- The amount of energy is either zero or a value given by the water-fill equation for $g \in \mathcal{G}^*$.

$$\mathcal{E}_{x,g} = \begin{cases} \Gamma \cdot \frac{2^{<b>}}{g_{geo}} - \frac{\Gamma}{g} & g > \frac{\Gamma}{K_{ma}} \\ 0 & g \leq \frac{\Gamma}{K_{ma}} \end{cases}$$

$$\text{transmit if } g > g_0 = \frac{\Gamma}{K_{ma}} .$$

- Direct water-fill calc from time-domain $g$ values is non-causal; can't really be made causal with delay.
- Instead, the stationary statistics have been exploited above.
  - Those statistics ($p_g$) may need to be estimated by counting $g$ values though over time.
  - If the system is ergodic, this will get better and better.
- Instantaneous transmit energy often has limit (less than the $\mathcal{E}_{x,g}$ value above) for small $g$.

**If so, reduce $\langle b \rangle$ -- this is a kind of margin also**

This is not practical – use erasures instead, but provides bound.

# Average and Outage Capacities

- **Average Capacity** is $\langle \tilde{C} \rangle = \sum_g p_g \cdot \log_2\left(1 + \mathcal{E}_{x,g} \cdot g\right)$ bits/complex-subsymbol.
  - $\langle \tilde{C} \rangle$ depends on energy distribution, which would be ergodic water-fill for maximum value.

$$= \log_2 \prod_{g \in \mathcal{G}^*} \left( \frac{K_{ma} \cdot g}{\Gamma} \right)^{p_g}$$

- **Outage Capacity** is $\tilde{C}_{out} \triangleq (1 - P_{out}) \cdot \langle \tilde{C} \rangle$ bits/complex-subsymbol.
  - Basically it reduces by data that would have been transmitted during outage
  - This may need retransmission, so would then be lower yet by average number of retransmissions + 1.

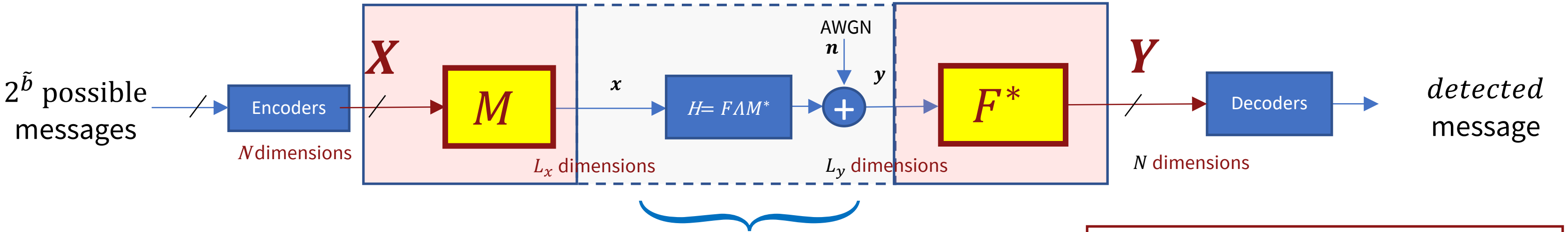
# Spatial Modulation

## “Space-Time Block Codes (STBC)”

# Spatial Vector Coding is Optimal

- The symbols have crosstalk or inter-spatial-dimension interference.
- However, symbols otherwise have no intersymbol interference, $\nu = 0$ (or ISI is separately handled).
  - With Vector DMT/OFDM on all crosstalking channels, there is no ISI.

- Spatial Vector-Code channel partitioning remains for each tone $n$ (index not shown).




$$y_l = \lambda_l \cdot x_l + n_l$$

- Parallel spatial channels index as:
  - $l = 1, \ldots, L \leq \wp_H \leq \min(L_x, L_y)$.

$$SNR_l = \frac{\lambda_l^2 \cdot \bar{\mathcal{E}}_l}{\sigma^2}$$

$$\mathbb{E}[\boldsymbol{n} \cdot \boldsymbol{n}^*] = R_{\boldsymbol{nn}} = R_{\boldsymbol{nn}}^{1/2} \cdot R_{\boldsymbol{nn}}^{*/2}$$

Noise-Equivalent Channel

$$\boldsymbol{y} \leftarrow R_{nn}^{-1/2} \boldsymbol{y} = \left( R_{nn}^{-1/2} \cdot H \right) \cdot \boldsymbol{x} + \tilde{\boldsymbol{n}}$$

# Spatial Modulator (Matrix)

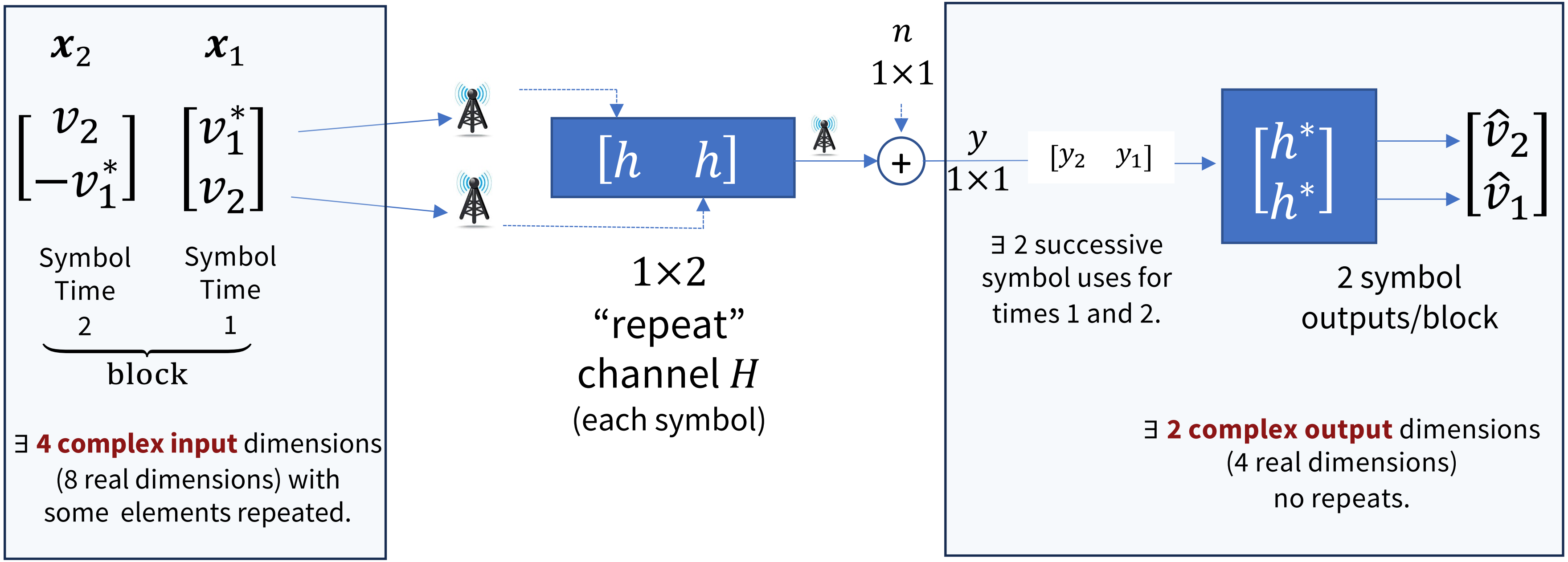
- Best matrix modulator is $M$ for any energy distribution.

- For this choice of $M$, it follows easily that the unbiased MMSE receiver matrix is $F^*$.

- In practice, it may be difficult for the transmitter to know $M$, since only the receiver can measure it.
  - This requires a reverse control channel.
  - The channel may change by the time it is reversed communicated.

- This leads to a variety of spatial approximations, among them Space-Time Block Codes (STBC).

# Alamouti's Code (1998)




- **The trivial 1 x 2 "repeat" channel is the ONLY channel** for which Alamouti's code is Vector Coding.
  - It obviously does better by 3 dB than a single channel use $2 \cdot [h]^2$ instead of $[h]^2$
- Basically, two line-of-sight paths to the same single-antenna receiver that must have the same gain.

# Alamouti with more general 2x2 H ??

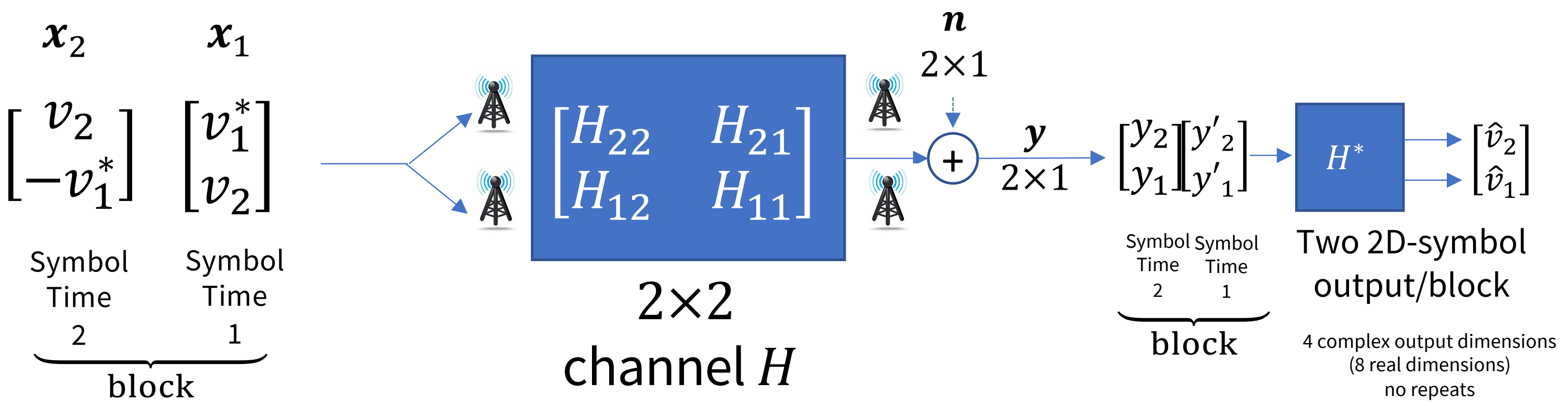
- Symmetric matrices: $J_2 \triangleq \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}$ $J_1 \triangleq \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$ $R \triangleq \begin{bmatrix} a & c \\ c & b \end{bmatrix}$
- Identity: $J_2 \cdot R \cdot J_2 + J_1 \cdot R \cdot J_1 = \begin{bmatrix} a+b & 0 \\ 0 & a+b \end{bmatrix}$

- Alamouti modulator: $\underbrace{\boldsymbol{x}}_{4\times 1} = \underbrace{\begin{bmatrix} J_2 \\ J_1 \end{bmatrix}}_{4\times 2} \cdot \underbrace{\begin{bmatrix} v_2 \\ v_1 \end{bmatrix}}_{\boldsymbol{v},2\times 1}$ $\boldsymbol{y} = \underbrace{\begin{bmatrix} H & 0 \\ 0 & H \end{bmatrix}}_{\boldsymbol{H}} \cdot \boldsymbol{x} + \boldsymbol{n} = \begin{bmatrix} H \cdot J_2 \\ H \cdot J_1 \end{bmatrix} \cdot \boldsymbol{v} + \boldsymbol{n}$

- Forward channel autocorrelation is the block diagonal $\boldsymbol{R_f} = \boldsymbol{H}^* \cdot \boldsymbol{H} = \begin{bmatrix} R_f & 0 \\ 0 & R_f \end{bmatrix}$.
- Each 2×2 sub-block diagonal is

$$\begin{aligned} R_f &= J_2^* \cdot H^* \cdot H \cdot J_2 + J_1 \cdot H^* \cdot H \cdot J_1 \\ &= \begin{bmatrix} H_{22}^2 + H_{21}^2 + H_{12}^2 + H_{11}^2 & 0 \\ 0 & H_{22}^2 + H_{21}^2 + H_{12}^2 + H_{11}^2 \end{bmatrix} \end{aligned}$$

# SUBOPTIMAL Alamouti's Code Use, 2 x 2 channel




4 complex input dimensions
(8 real dimensions)
some symbol values repeated

$$
\begin{aligned}
R_f &= J_2^* \cdot H^* \cdot H \cdot J_2 + J_1 \cdot H^* \cdot H \cdot J_1 \\
&= \begin{bmatrix} H_{22}^2 + H_{21}^2 + H_{12}^2 + H_{11}^2 & 0 \\ 0 & H_{22}^2 + H_{21}^2 + H_{12}^2 + H_{11}^2 \end{bmatrix}
\end{aligned}
$$

$$\gamma_{repeat} = \frac{1}{2}$$

- The transmitter remains channel independent, and the receiver remains simple matched matrix.
- The SNR is higher, BUT the data rate is halved. **Vector coding is better.**

# STBC Codebooks

- Variety of different sizes, but rate loss $\gamma_{repeat} \geq$ ½ (with more than one receive antenna).
- They perform a lot worse than vector-coding in practice.

Example: $\bar{\mathcal{E}}_x = 1\,;\ \sigma^2 = 1$

```
>> H
H =
    1.0000    0.9000
    0.8000    1.0000
>> [F,L,M]=svd(H)
F =
   -0.7245   -0.6892
   -0.6892    0.7245
L =
    1.8512         0
         0    0.1512
M =
   -0.6892   -0.7245
   -0.7245    0.6892
bvc=log2(det(L^2+eye(2))) = 2.1790
bstbc=log2(sqrt(norm(H,'fro')^2+1)) = 1.0769
10*log10((2^bvc-1)/(2^bstbc - 1))  =   5.0245 dB!
```

Sqrt same as ½ in front, must include the repeat Factor on the two inputs.

**At higher data rates, the VC advantage grows;**

**Wireless Raleigh fading**

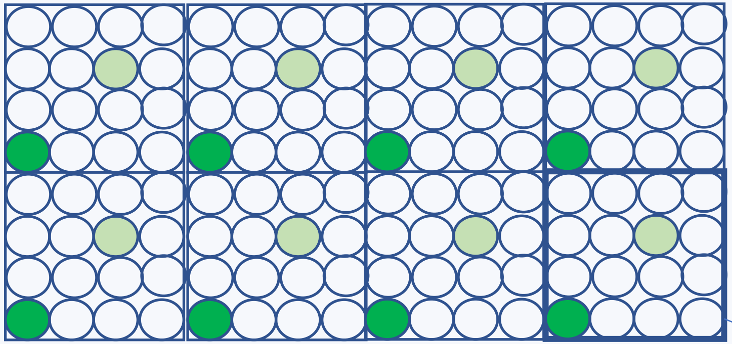
$$\langle P_e \rangle \propto (SNR)^{d_{free}}$$

**but still large VC advantage.**

# Wireless field use has STBC as “option”

- STBC will essentially repeat symbols, so if the channel is very poor and the codes (nonideal) are fixed, then it can increase SNR and create a reliable link at significant data-rate loss.
- This may be acceptable if no connection is otherwise reliably possible.
- For ideal outer codes ($\Gamma = 0$ dB), there is no such repeat-energy advantage theoretically.
- Most wireless systems (Cellular and Wi-Fi) allow use of STBC as an option, but also permit use of vector-coding approximations to avoid the STBC losses.

# Cellular: Type 1 Precoders



$L_{x,2} \cdot O_2$ choices

$L_{x,1} \cdot O_1$ choices

$O_i$ is oversampling factor (angle)

$L_x = L_{x,1} \cdot L_{x,2}$ = number used antennas

Each $O_1 \cdot O_2$ rectangle is a "subarray" of choices.

$Q_{L_{x,1} \cdot O_1} \otimes Q_{L_{x,2} \cdot O_2}$ = precoder factor is cartesian product of horizontal and vertical DFTs

$Q_{L_{x,i} \cdot O_i}$ is a DFT of size in subscript

- The precoder matrix $A$ stacks the columns of $Q_{L_{x,1} \cdot O_1} \otimes Q_{L_{x,2} \cdot O_2}$ so each is $L_x$ x 1
- These columns can be multiplied by $1, j, -1, -j$ for different streams to form an $L_x$ x ss matrix

# Cellular: Type 2 Precoders

- Use the columns of Type 1, call them $\boldsymbol{w}_{l,ss}$ and weight them (not necessarily unity gain).

- Weights $a_{l,ss}$ can have amplitudes $2^{-i/2}$ for $i \in \{0, \ldots, 6\}$ and phases $\frac{2\pi}{j}$ $j \in \{0, \ldots, L_x - 1\}$.

$$W_{ss} = \sum_{\ell=1}^{L_x} \boldsymbol{w}_{\ell,ss} \cdot a_{\ell,ss}$$

- Receiver will also send back indices $i$ and $j$ along with indices for the Type 1 $\boldsymbol{w}_{l,ss}$ .

- Factors $a_{l,ss}$ are often split into wideband slower-varying factor and narrow band (frequency-dependent) faster-varying factor.

# Wi-Fi Spatial Modulators

- Wi-Fi Space-Time is much simpler than cellular.
  - But alas, also allows Alamouti on 2x2 antenna mix channels,
    - and will repeat them for several 2x2 groupings (ouch!).

- Specifies a series of 2x2 rotations (complex, 2 angles) and the spatial antenna indices to which they apply to attempt to approximate $M$ (or exactly realize it if there are enough sent).
  - That's more like it!!

# End Lecture 5